

# Type-Aware Error Control for Robust Interactive Video Services over Time-Varying Wireless Channels

Wei-Kuo Liao, *Member, IEEE*, and Yi-Hsuan Lai

**Abstract**—We propose a scheme called Type-Aware Error Control (TAEC) to provide robust interactive video services over time-varying wireless channels. Basically, TAEC explores the temporal dependence of video frames by pushing out the lower impact frames. In doing so, queuing delay is controlled and the important frame is retransmitted multiple times. We consider the application of the nonstationary stochastic control to further improve the wireless system throughput by exploring the multiuser diversity gain. The techniques involved in our proposed solution consist of constructing a Markov model, reducing the number of parameters estimated, and avoiding the state explosion problem. The provisioning of immediate bandwidth guarantee is another feature of TAEC, i.e., when exploring the multiuser diversity gain, the users share their reserved bandwidth only if the perceived video qualities are acceptable. As shown in the simulation, TAEC improves the perceived video quality in terms of luminance PSNR when compared to other schemes.

**Index Terms**—Video communication, real-time scheduling, wireless error control, nonstationary stochastic control.

## 1 INTRODUCTION

WITH advances in wireless technology, provisioning of cost-effective interactive video service via wireless access networks, e.g., WLAN and WiMAX, is still a challenge. The key issue is that wireless channels are error-prone. Thus, delivering video streams over a wireless channel results in frequent packet losses. Coded video streams, in particular, are very vulnerable to packet loss. Therefore, even if the wireless network is equipped with a packet scheduling algorithm which is extensively explored for wireline networks to provide bandwidth guarantees, the perceived video quality can still be seriously degraded.

This study aims to develop an effective retransmission-based error control for wireless channels to reduce video quality loss. The retransmission is very primitive in form. It is also efficient, because in order to repair error, a packet is retransmitted only if its acknowledgment is not received in time. However, when adopting the retransmission-based error control over wireless channels for video communication, we identify three issues that remain to be addressed: 1) the number of times a packet can be retransmitted, 2) which packet is discarded or allowed to be retransmitted multiple times, and 3) when and how the *multiuser diversity gain*, i.e., the reserved bandwidth of a connection to be shared by other connections to improve the performance of the overall system, will be explored.

The first issue mentioned above is raised because packet retransmission delays the transmission of succeeding packets in the same connection. Such delay propagation may lead to the dissatisfaction of the timing requirement of interactive video services and thus must be properly restricted. We have to address the second issue due to the temporal dependence of video frames within a group of successive frames called group of pictures (GoP) in a coded video stream. In a GoP, each frame is associated with a type specifying how it depends on the other frames in the same GoP for reconstructing the image. Apparently, the quality impact of a packet is characterized by the type of carried frame and the loss status of the frames in the same GoP. Therefore, to improve video quality, it is plausible that the lower impact packets might be discarded earlier to allow other important packets to be retransmitted if necessary. Finally, rational users might only have incentive to share their reserved bandwidth with others when their perceived video qualities are acceptable. In such a case, we can say that the users require an *immediate bandwidth guarantee*. To the best of our knowledge, no previous studies have focused on maximizing the performance of a wireless system subject to such a bandwidth guarantee.

Knowing this, we propose a method called type-aware error control (TAEC). Broadly speaking, TAEC consists of two major components, namely the transport architecture and the self-tuning controller. In transport architecture, TAEC assumes the minimum bandwidth reservation admitted to each connection and adopts a packet scheduling algorithm for the wireline network. Accordingly, a feasible polling sequence is generated to specify the order to grant (or poll) the connections for transmitting their packets. Upon the reception of a poll for a connection, TAEC may push out zero or more video frames in the per-connection buffer and then deliver the packet of the first video frame in the buffer. Due to the pushout style (also called drop-from-front or drophead), the timing requirement of video frames can still

• W.-K. Liao is with the Department of Electrical Engineering, National Chiao Tung University, 1001 University Road, Hsinchu, Taiwan 300, R.O.C. E-mail: liaoweik@gmail.com.

• Y.-H. Lai is with the Department of Applied Mathematics, Hsuan Chuang University, 48, Hsuan Chuang Road, Hsinchu City, Taiwan 300, R.O.C. E-mail: laiyhhsuan@wmail.hcu.edu.tw.

Manuscript received 6 June 2008; revised 14 Apr. 2009; accepted 28 Feb. 2010; published online 22 July 2010.

For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-2008-06-0219. Digital Object Identifier no. 10.1109/TMC.2010.141.

be firmly guaranteed merely by limiting the buffer size. Moreover, the order of frames can be preserved to ease the receiver's workload.

The self-tuning controller employed in TAEC aims at increasing the number of received video frames, which can be fully reconstructed. When a poll is received, according to the types of current frames buffered in the base station, the channel condition, and other historical information, the controller is based on the nonstationary stochastic control theory (a survey on this topic can be found in [7]) to determine how many lower impact frames need to be pushed out. Furthermore, to explore the multiuser diversity gain subject to the immediate bandwidth guarantee, we propose a method rooted in *one-step policy improvement* [22] for the controller to decide how to alter the original polling sequence to improve the overall quality of the system. Unlike a related study [22], the proposed method is in a more relaxed form and is applicable in a nonstationary stochastic environment. In operation, a set of initial policies and corresponding values is generated offline and downloaded into a table a priori. The base station then decides and performs the improved error control action by looking up the table and finding the minimum value.

In short, TAEC attempts to maximize the number of received video frames, which can be fully reconstructed, subject to the immediate bandwidth guarantee in anticipation that the overall received video quality can be sustained. We evaluate the TAEC by simulating coded video streams through an IEEE 802.11 wireless network. As shown in the simulation, the performance of the TAEC is fairly good in terms of peak signal-to-noise ratio (PSNR) when compared to other schemes.

The remainder of this paper is organized as follows: A brief survey of related work is given in Section 2. Section 3 gives the system specifications. In Section 4, we describe our system by using a Markov chain model. Section 5 gives procedures to generate initial and improved policies. Simulation results are displayed in Section 6. We draw conclusions in Section 7. The Appendix contains the theoretical foundations for our study.

## 2 RELATED WORK

Retransmitting selective packets for interactive service has been long investigated since this method incurs less bandwidth overhead. When considering the end-to-end connection, it is noted that a frame might be late after retransmission due to the lengthy round-trip delay. However, such late-arrived frames can still be used to construct subsequent frames. Seeing this, Rhee [15] proposed a novel selective retransmission technique to exploit such benefits. In the proposed technique, every  $p$ th video frame is treated as an important one. Each frame can only temporally depend on the previous one, except for the important frame, which can only temporally depend on the previous important frame. Moreover, only the packet carrying the important frame can be retransmitted. Combined with layered coding, the technique was shown to be effective even under heavy packet loss due to network congestion. On the other hand, Yu et al. [21] considered an end-to-end scenario which consisted of wireless access links in both ends. They proposed to place proxies in the base stations, and suggested that along the end-to-end path, the retransmission could

only take place locally in the downlink transmission. In addition, retransmission was only allowed if the frame was not late and the succeeding frame had not yet arrived at the base station. As such, the delay caused by the retransmission could be restricted. For wireless channels, Chebrolu and Rao [4] and Aramvith et al. [2] proposed to conditionally retransmit the important frames. A similar idea was also applied to voice transmission by Sanneck et al. [16]. Their method allowed for one retransmission of the important packets. Their proposed schemes were shown to be effective by simulations.

When exploring multiuser diversity gain, an intuitive way is to let the connection with better channel condition transmit. By doing so, the utility of wireless bandwidth can be improved. Following such intuition, earlier works, e.g., *opportunistic scheduling* by Liu et al. [14], were proposed to maximize the system-utility subject to long-term quality-of-service guarantees for connections. However, due to the temporal correlation in the coded video stream, video quality is degraded more seriously by the loss of an important frame. To resolve this, Typelly et al. [19] proposed a scheduling scheme which simply assigned the connection with priority for scheduling as the product of the importance of the *head-of-line* frame (i.e., the first frame in the buffer), channel condition, etc. We thus refer to this scheme as the *multiplication of priorities*. A similar idea has been applied by Tang et al. [18], where interframe delay was also considered as a factor in the multiplication of priorities. However, it is hard to justify whether or not such schemes can effectively improve overall throughput.

In deriving a model for self-tuning controller, we make several assumptions, e.g., the Rayleigh-fading channel, to make our model Markovian (see [23], for the fitness of Markov chain to describe the time-varying behavior of received signal strength (RSS) over a Rayleigh-fading channel. However, the assumptions are not realistic in practice. For example, studies [6], [17] have claimed that it is better to use the semi-Markov process, rather than a time-homogeneous Markov chain, for protocol evaluation in indoor environments with frequency-selective channels. Therefore, the nonstationary stochastic control is applied in TAEC to cope with the inevitable modeling error and nonstationary parameter variation problem.

## 3 SYSTEM SPECIFICATION

We consider a frequency duplex time division system, i.e., two common-frequency bands for uplink and downlink transmissions, respectively, shared by multiple users/connections. The access to each common band is coordinated by the base station through polling. An immediate acknowledgment encoding the transmission outcome (success or failed) and detected channel status is sent back to the sender upon the reception of a packet. We assume that the acknowledgment is always received by the sender. After being successfully transmitted, a packet is automatically removed from its *per-connection buffer*, or simply its buffer.

Each connection is associated with a reserved bandwidth. A polling algorithm runs in the base station to advise when to poll a connection to inform it of delivering a packet immediately. In downlink transmission, the base station checks the advice from the polling algorithm and the current system state to dictate which connection to poll.

Let  $L$  be the maximum number of frames stored in a buffer. Upon the arrival of a frame, the head-of-line frame is pushed out if overflow occurs. At a specified instant, referred to as *evaluation time*, the transmitter may perform the *admissible action*. The admissible action is either to deliver nothing, or to first push out  $j$  packets. Then, the head-of-line frame is encapsulated into a packet and delivered over the channel. The general action space is  $\{-1, 0, 1, \dots, \max\{0, \ell - 1\}\}$ , where  $-1$  indicates delivering nothing and  $\ell$  is the length of current buffer occupancy. As such, the queuing delay of any successfully transmitted packet is no more than  $L$  packet interarrival times, regardless of the action being taken.

## 4 MARKOV DECISION MODEL AND ASSUMPTIONS

This section provides a brief description of the time-homogeneous Markov decision process and the formulation of the system under study.

Let  $\mathbb{X}$  be a finite state space and  $c : \mathbb{X} \times \mathbb{X} \rightarrow [0, \infty)$  be a bounded state-transition cost function. In addition, let  $X_i$  be the state at the  $i$ th timepoint with  $X_i \in \mathbb{X}$ . The criterion to be considered is to minimize the average cost over infinite horizon with the objective function specified as follows:

$$\limsup_{t \rightarrow \infty} \frac{\sum_{i=1}^t c(X_i, X_{i+1})}{t} \text{ almost surely,} \quad (1)$$

where the ‘‘almost surely’’ is w.r.t. probability measure that depends on the policy, which will be introduced immediately, and thus the minimization is over the policies.

Denote  $\mathcal{U}(x)$  as the admissible action set under state  $x$ . A *stationary policy* is a function from a state, say  $x$ , to a distribution over admissible actions of the state, i.e.,  $\mathcal{U}(x)$ . A *deterministic policy* is a function of a state to an action in the admissible action set of the state. With the given criterion, when dealing with the time-homogeneous Markov chain with finitely many states and actions, the existence of an optimal deterministic policy is well known.

One of the principal methods used to find the optimal deterministic policy is the *policy iterative procedure* [9] which iterates between the *value determination step* and the *policy improvement step*. Given a policy  $\pi$ , let  $P_{xy}^\pi$  be the transition probability from  $x$  to  $y$  under the action  $\pi(x)$ . Consider the following *Poisson’s equation* under a given policy  $\pi$ :

$$v^\pi(x) + \eta_\pi = \sum_y P_{xy}^\pi (c(x, y) + v^\pi(y)), \quad \forall x \in \mathbb{X},$$

where  $(v^\pi(\cdot), \eta_\pi)$  is the solution to the Poisson’s equation. We will refer to  $v^\pi(x)$  as the *relative cost* of state  $x$ . Moreover,  $\eta_\pi$  is the expected average cost. In the value determination step, we find the solution  $(v^\pi(\cdot), \eta_\pi)$  to the Poisson’s equation under policy  $\pi$ . It can be shown that if  $c(\cdot, \cdot)$  is bounded, then  $v^\pi(\cdot)$  and  $\eta_\pi$  are bounded. To ease our discussion, we introduce the notation  $Q_{P^u}[c(X, Y) + v(Y)|X]$  which is a conditional expectation defined as follows:

$$Q_{P^u}[c(X, Y) + v(Y)|X = x] := \sum_y P_{xy}^u (c(x, y) + v(y)), \quad (2)$$

where  $P_{xy}^u$  is the transition probability from  $x$  to  $y$  with respect to the probability under the action  $u$ . The policy improvement step, which basically decomposes the

optimization problem into per-state subproblems, is listed as follows:

$$\arg \min_{u \in \mathcal{U}(x)} Q_{P^u}[c(X, Y) + v(Y)|X = x], \quad \forall x \in \mathbb{X}. \quad (3)$$

Denote  $\pi'(x)$  as the minimizer for the per-state subproblem with respect to state  $x$ . Then,  $\pi'$  is the improved policy which becomes the given policy in the next iteration.

### 4.1 Defining State

To explore the temporal correlation between frames, we define a notation called *loss indication*, denoted by  $W$ , which takes values of  $\{0, 1\}$ . The loss indication captures whether the reference frame of the head-of-line frame is lost or not.

To keep the number of states in the model to a manageable size, we consider the simple two-state Markov chain to model the channel state. Let  $\mathcal{CH}_t$  be the channel state at time  $t$ , which is either  $g$  (good) or  $b$  (bad). Note that since channel state  $\mathcal{CH}_t$  is not observable at time  $t$ , we then denote the most recent observable channel state at time  $t$  as  $Ch_t$  which is channel condition  $\mathcal{CH}_{t^-}$  where  $t^-$  is the time when the last acknowledgment prior to time  $t$  was received.

In summary, the basic state at time  $t$  is thus defined as

$$X_t := (Ch_t, W_t, B_t^1, B_t^2, B_t^3, \dots, B_t^L) = (Ch_t, W_t, \mathbb{B}_t),$$

where  $W_t$  is the loss indication,  $L$  is the buffer size,  $B_t^i$  is the type of frame (or colors of frame) in buffer location  $i$ , and  $\mathbb{B}_t$  is a shorthand of  $B_t^1, B_t^2, B_t^3, \dots, B_t^L$  which is referred to as backlogged status. The  $B_t^i$  take values of  $\{-1\} \cup M$  where  $-1$  indicates no frame and  $M$  is the set of frame types.

### 4.2 Transition Rules and Assumptions

Recall that we always update the state at an evaluation time. Accordingly, the state jumps only at these evaluation times. Let the current time be  $t$  at which the state is being evaluated and the next evaluation time be  $t + \tau$  with some positive integer  $\tau$ . We always let  $Ch_{t'} = Ch_{t+1}$  for all  $t' \in [t + 1, t + \tau)$ , i.e., the channel state remains the same from  $t + 1$  to the next evaluation time of the connection. In the case that we deliver a packet at the evaluation time  $t + \tau$ ,  $Ch_{t+\tau+1} = \mathcal{CH}_t$ .

Process  $\mathcal{CH}_t$  is assumed to preserve the Markovian property. Furthermore, we assume that transmission outcome  $K_t$  with  $K_t \in \{s(\text{success}), f(\text{failure})\}$ , is independent of other historical information conditioning on  $\mathcal{CH}_t$ , namely the *assumption of outcome*. Denote the sequence of video frame arrivals accumulated in  $[t, t + \tau - 1]$  as  $A(t, \tau)$ , which is assumed to be independent of history prior to  $t$  conditioning on  $X_t$ , namely the *assumption of arrival*. Denote the action performed at time  $t$  as  $u_t$ . This study assumes that given the current backlogged status  $\mathbb{B}_t$ ,  $A(t, \tau)$ , transmission outcome  $K_t$ , and  $u_t$ , we can derive the resultant backlogged status  $\mathbb{B}_{t+\tau}$ .

Let  $x$  be a state and  $x_{ch}$ ,  $x_w$ , and  $x_b$  represent the channel state, loss indication, and backlogged status components in  $x$ , respectively. Let  $P$  be the probability under the policy sequence  $\{\pi_t\}$ . By convention, we always let  $P(a|b) = 0$  if  $P(b) = 0$ . We then derive the transition probability per each evaluation by repeatedly using the conditional probability rule  $P\{a, b|c\} = P\{a|b, c\} \times P\{b|c\}$  as follows:

$$\begin{aligned}
& P\{X_{t+\tau} = x | X_t, X_{t-1}, X_{t-2}, \dots\} \\
&= P\{Ch_{t+\tau} = x_{ch}, W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b | \\
&\quad Ch_t, W_t, \mathbb{B}_t, X_{t-1}, X_{t-2}, \dots\} \\
&= P\{W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b | \\
&\quad Ch_{t+\tau} = x_{ch}, Ch_t, W_t, \mathbb{B}_t, X_{t-1}, X_{t-2}, \dots\} \\
&\quad \times P\{Ch_{t+\tau} = x_{ch} | Ch_t, W_t, \mathbb{B}_t, X_{t-1}, X_{t-2}, \dots\} \\
&= P\{Ch_{t+\tau} = x_{ch} | Ch_t\} \\
&\quad \times P\{W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b | \\
&\quad Ch_{t+\tau} = x_{ch}, W_t, \mathbb{B}_t, X_{t-1}, X_{t-2}, \dots\}.
\end{aligned}$$

When we take into account the transmission outcome and packet arrival, the transition probability of basic states can be rewritten as follows:

$$\begin{aligned}
& P\{X_{t+\tau} = x | X_t, X_{t-1}, X_{t-2}, \dots\} \\
&= \sum_{k \in \{s, f\}} P\{Ch_{t+\tau} = x_{ch} | Ch_t\} \\
&\quad \times P\{W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b, K_t = k | \\
&\quad Ch_{t+\tau} = x_{ch}, W_t, \mathbb{B}_t, X_{t-1}, X_{t-2}, \dots\} \\
&= \sum_{k \in \{s, f\}} P\{Ch_{t+\tau} = x_{ch} | Ch_t\} \\
&\quad \times P\{K_t = k | Ch_{t+\tau} = x_{ch}\} \\
&\quad \times P\{W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b | \\
&\quad Ch_{t+\tau} = x_{ch}, W_t, \mathbb{B}_t, K_t = k, X_{t-1}, X_{t-2}, \dots\} \\
&= \sum_{\xi \in \mathcal{A}(X_t)} \sum_{k \in \{s, f\}} P\{Ch_{t+\tau} = x_{ch} | Ch_t\} \\
&\quad \times P\{K_t = k | Ch_{t+\tau} = x_{ch}\} \\
&\quad \times P\{W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b | \\
&\quad A(t, \tau) = \xi, W_t, \mathbb{B}_t, K_t = k\} \times P\{A(t, \tau) = \xi\} \\
&= P\{X_{t+\tau} = x | X_t\},
\end{aligned} \tag{4}$$

where the second equality is obtained by applying the conditional probability rule and assumption of outcome, and the third equality is due to the application of the conditional probability rule and assumption of arrival. Moreover,  $\mathcal{A}(x)$  denotes all the possible sequences of video frame arrivals between two consecutive evaluation times with current state  $x$ . The derivation also concludes that  $X_{t+\tau}$  is conditionally independent of  $X_{t-1}, X_{t-2}, \dots$  given  $X_t$ , and thus we can claim that our system is Markovian. Note that  $P\{W_{t+\tau} = x_w, \mathbb{B}_{t+\tau} = x_b | A(t, \tau) = \xi, W_t, \mathbb{B}_t, K_t = k\}$  is either zero or one. In this study, we only consider cases with current state  $x$  and the number of video frame arrivals as able to derive the sequence of video frame arrivals.

The above model is mostly frame-level, i.e., each element in the probability space is a function of video frame arrival sequence, video frame transmission outcome sequence, and channel condition sequence. In practice, a large video frame is usually segmented into several packets according to the maximum segment size. The packet-based scheduler then treats each packet as an individual entity and separately schedules each one to transmit in order. As an immediate consequence, the probability of successfully transmitting a whole video frame depends on how many packets of the frame have been successfully delivered, e.g., such probability increases when more packets of the frame have been successfully delivered. However, when model complexity becomes a concern, we may not be able to capture the detailed information completely. One way to reduce the model complexity is by pretending that the delivery is

transmitting the whole frame after delivering a packet. Upon reception of the acknowledgment, we update the channel condition as usual. Furthermore, if the packet is successfully delivered, it is removed from the head-of-line of the buffer; otherwise it remains in the buffer. Till the delivery of the last packet of the frame succeeds, we claim that the frame has been successfully transmitted. Otherwise, the transmission outcome of the frame remains a failure even when some of its packets have been successfully delivered. When pushing out the frame, all the packets of the frame are pushed out.

With the above procedure for model complexity reduction in place, suppose that the frame is segmented into  $\nu$  packets. What we have to redefine is simply the probability  $P\{K_t = \text{success} | Ch_{t+\tau} = x_{ch}\}$ , namely the *aggregated success probability*, as the original one divided by  $\nu$ , and the *aggregated failure probability*  $P\{K_t = \text{failure} | Ch_{t+\tau} = x_{ch}\} = 1 - P\{K_t = \text{success} | Ch_{t+\tau} = x_{ch}\}$ . Though inducing extra modeling error, such an aggregated model remains just about as frame-level as the original one. In the following section, we will discuss how to instruct the packet-based scheduler to use information provided by the aggregated model.

### 4.3 Definition of Cost and Conditional Expectation

Define

$$\begin{aligned}
H(X_t, \xi, k) &:= \{y = (y_{ch}, y_w, y_b) : P\{W_{t+\tau} = y_w, \mathbb{B}_{t+\tau} \\
&= y_b | A(t, \tau) = \xi, W_t, \mathbb{B}_t, K_t = k\} = 1\}.
\end{aligned}$$

To save the notation, we denote  $\xi$  as either the number of frame arrivals or the sequence of frame arrivals, if without ambiguity. Let  $R(X_t) := \bigcup_{(\xi, k) \in \{s, f\} \times \{0, 1, 2\}} H(X_t, \xi, k)$ . In addition, let

$$\begin{aligned}
K(X_t, \xi, y) &:= \{k : P\{W_{t+\tau} = y_w, \mathbb{B}_{t+\tau} = y_b | A(t, \tau) \\
&= \xi, W_t, \mathbb{B}_t, K_t = k\} = 1\}.
\end{aligned}$$

Therefore, the conditional expectation, which will be exploited for deriving the policy in the next section can be written as follows:

$$\begin{aligned}
& \mathbf{E}[c_t(X_t, X_{t+\tau}) + v_{t+\tau}(X_{t+\tau}) | X_t] \\
&= \sum_{y \in R(X_t)} \sum_{\xi \in \{0, 1, 2\}} \sum_{k \in K(X_t, \xi, y)} P\{Ch_{t+\tau} = y_{ch} | Ch_t\} \\
&\quad \times P\{K_t = k | Ch_{t+\tau} = y_{ch}\} \\
&\quad \times P\{A(t, \tau) = \xi\} \times (c'(X_t, \xi, k, \pi_t(X_t)) + v(y)),
\end{aligned} \tag{5}$$

where  $c'(X_t, \xi, k, u)$  is the number of frames pushed out by action  $u$  and the number of frames being pushed out due to overflow between  $t$  and  $t + \tau$  given that the state at time  $t$  is  $X_t$ , the transmission outcome at time  $t$  is  $k$ , and the number of frame arrivals during the period from  $t$  to  $t + \tau$  is  $\xi$ . It is obvious that  $c_t$  is also an expectation. The above form of conditional expectation is for the case in which  $u_t$  delivers the head-of-line frame. We can obtain this for cases where  $u_t$  does not deliver any frame in a similar way and thus we omit the derivation.

## 5 POLICY DERIVATION

The derivation of a policy to explore multiuser diversity gain is described in this section. Basically, methods **M1** and **M2**, as shown in the Appendix, are applied to do so. The

application of **M1** derives an initial policy, which is a stepping stone for applying **M2** to obtain the final policy, or so-called improved policy.

### 5.1 Analysis

Suppose there are  $N$  connections. Denote  $X_t^n$  as the state of connection  $n$  at time  $t$ . Let the *original schedule* be the schedule generated by the poller. The controller selects a connection to poll based on the original schedule and current system state. If the controller simply follows the advice from the poller, then only the connection to be polled is being evaluated. Otherwise, both the connection to be polled and the connection advised to be polled by the original schedule are evaluated. The transition cost of individual connection is calculated during the evaluation; otherwise, the transition cost is zero.

Denote  $X_t$  by the overall system state  $(X_t^1, X_t^2, \dots, X_t^N)$ . Let  $\mathbb{X}^n$  be the set where  $X_t^n$  takes the value and let  $\mathbb{X}$  be  $\prod_{1 \leq n \leq N} \mathbb{X}^n$ . The summation of the transition costs experienced by all the connections at the starting time of a transmission is then defined as the *transition cost of the system* at that time.

When exploring multiuser diversity gain, we might apply the policy iterative procedure to obtain the optimal policy. However, in doing so, the number of unknowns becomes enormous if  $N > 1$ , e.g.,  $h^N$  if the number of states for each connection is  $h$ . Solving the system of equations by inverting a matrix in value determination step is computationally prohibitive. To avoid such a large-scale computation, we consider using method M2. The procedure includes finding a simple *initial policy* which is not good enough but could ease the computation of value determination.

### 5.2 Derivation of Initial Policy

To meet the requirement of simplicity, we consider letting each connection make its own decision without altering the original schedule in the initial policy. Therefore, the initial policy is obtained by applying the policy iterative procedure for each connection independently. In doing so, factors contributing to the transition probabilities, e.g., the parameters of the bit error rate in each channel state, the transition probabilities of channel states, and packet arrival rates all need to be estimated. However, the update of each parameter results in the change of transition probability and thus leads to the reexecution of the policy improvement routine. This raises the practical concern that computing policy on the fly maybe too computationally intensive.

Other than the aforementioned estimation problem, the nonstationary effect on channel condition could be significant within a time scale as small as an interpolling interval. In addition, the polling algorithm itself may also result in nonstationarity. Specifically, the perfect periodic polling proposed in [3] usually cannot achieve 100 percent utilization. Therefore, we consider using a well-known polling method called *periodic poller* [13] or *deadline-driven poller* to avoid possible confusion that results with perfectly periodic polling. Here is its basic working principle: Denote the deadline for the connection  $n$  by  $d_n$ . In addition, the connection is said to be *ready* if  $d_n - \phi_n$  is not greater than the current time, where  $\phi_n$  has been defined as the requested

1. For all  $n$ ,  $d_n \leftarrow 0$ ;
2. poll the ready connection, say  $n$ , with the smallest deadline;
3.  $d_n \leftarrow d_n + \phi_n$ ;
4. wait till the completion of transmission pattern and goto 2.

Fig. 1. The deadline-driven polling.

period. The periodic poller then works as shown in Fig. 1. In such a polling method, all the ready connections are polled prior to its deadline plus a constant  $b$ , where  $b = \max_n e_n$ , if the total utilization  $\sum_n e_n / \phi_n \leq 1$ , i.e., we can ensure that a poll arrival for connection  $n$  must fall into the interval  $[d_n - \phi_n, d_n + b]$ . However, the interpolling time for connection  $n$  could be as small as  $e_n$  and as large as  $2\phi_n - 1 + b$ . The time-varying interpolling times then lead to the time-varying transition probabilities of channel state and thus the time-varying transition matrices of basic state. Worse yet, the original schedule will indeed be altered when multiuser diversity gain is explored. In such a case, the controller may skip several polling chances for a particular connection, and thus the variation of transition matrices could be further increased.

To handle the problems of estimating multiple parameters and non-time-homogeneous transition probabilities, we apply method **M1**, shown in the Appendix, as follows: Let  $t_i$  be the time of the  $i$ th evaluation of the connection under study. We configure the probabilities  $\hat{p}_{ch}(b|g)$ ,  $\hat{p}_{ch}(g|b)$ ,  $\hat{p}_{e,g}$  (the packet error rate in a good state),  $\hat{p}_{e,b}$  (the packet error rate in a bad state), and  $\hat{p}_a(1)$  ( $\lambda \hat{p}_a(1)$  is the frame arrival probability where  $\lambda$  is the parameter to be estimated by our proposed *maximum update rule* [12]). Let  $\lambda(t)$  denote the  $\lambda$  value at time  $t$  and let  $\pi^\lambda$  denote the policy corresponding to  $\lambda$ . Therefore, for cases where  $\pi^\lambda(x) \neq -1$  and it is the action to be taken, the conditional expectation in (5), rewritten as  $\mathbf{E}[c_{t_i}(X_{t_i}, X_{t_{i+1}}) + v_{t_{i+1}}^{\pi^\lambda(t_i)}(X_{t_{i+1}})|X_{t_i}]$ , is approximated as follows:

$$\begin{aligned} & Q_{\hat{p}_x^{\lambda(t_i)}(x)}[c(X, Y) + v^{\pi^\lambda(t_i)}(Y)|X = x] \\ & := \sum_{y \in \hat{R}(x)} \sum_{\xi \in \{0,1,2\}} \sum_{k \in \hat{K}(x, \xi, y)} \hat{p}_{ch}(y_{ch}|x_{ch}) \\ & \quad \times \hat{p}_K(k|y_{ch}) \hat{p}_A(\xi) \left( c'(x, \xi, k, \pi^\lambda(t_i)(x)) + v^{\pi^\lambda(t_i)}(y) \right), \end{aligned}$$

where  $\hat{R}(x)$  and  $\hat{K}(x, \xi, y)$  are defined as  $R(x)$  and  $K(x, \xi, y)$  according to the configured probability with respect to action  $\pi^\lambda(t_i)(x)$ , respectively. The above approximated value is denoted as  $\hat{Q}$ . For cases where the action to be taken at time  $t$  is  $-1$ , i.e., no delivery, the conditional expectation in (5) is approximated as follows:

$$\begin{aligned} & Q_{\hat{p}_{-1}}[c(X, Y) + v^{\pi^\lambda(t_i)}(Y)|X = x] \\ & := \sum_{y \in \hat{R}_{-1}(x)} \left( \sum_{\xi \in \{0,1,2\}, f \in \hat{K}(x, \xi, y)} \hat{p}_A(\xi) \left( c'(x, \xi, f, \pi^\lambda(t_i)(x)) + v^{\pi^\lambda(t_i)}(y) \right) \right), \end{aligned}$$

where the probabilities related to the channel are ignored because no packet is allowed to transmit over the channel. Furthermore,  $\hat{R}_{-1}(x)$  is defined as  $\hat{R}(x)$  but with respect to the action  $-1$ . For example, if  $y \in \hat{R}_{-1}(x)$ , then  $y_w$  must be the same as that in cases where the transmission outcome is a failure, and besides,  $y_{ch} = x_{ch}$  because no acknowledgment is received to update the channel state. The above approximated value is denoted as  $\hat{Q}_{-1}$ .

To find policy  $\pi^{\lambda(t)}$ , we quantize the range of  $\lambda$ . Based on each quantization value of  $\lambda$  and the configured probabilities, the problem is formulated as a time-homogeneous Markov decision problem with configured probabilities corresponding to  $\lambda$ . It can be proved that the corresponding Markov chain is aperiodic, with a single recurrent chain, if  $\hat{p}_a(0) > 0$ . Thus, we perform the policy iterative procedure to obtain the deterministic policy. The resultant policies are stored in a table. The controller then looks up the table by using the index of state and calibrated  $\lambda$  at time  $t$  to obtain the corresponding action.

### 5.3 Deriving Improved Policy

Let  $\hat{Q}_t^m$  and  $\hat{Q}_{-1,t}^m$  be the  $\hat{Q}$  and  $\hat{Q}_{-1}$  value of connection  $m$  at time  $t$ , respectively. The idea of deriving improved policy is that based on the relative values produced by the initial policy, we improve the policy by executing a policy improvement step in an online fashion to greedily minimize  $\Delta Z_t$  in anticipation that  $Z_t$  can consequently be decreased. Recall the definition of  $\Delta Z_t$ , as in the Appendix:

$$\Delta Z_t = \mathbf{E}[c_t(X_t, X_{t+1}) + v_{t+1}(X_{t+1}) | X_t] - v_t(X_t) - \eta_t.$$

Suppose that the poller advises connection  $m$  to transmit at time  $t$ . Note that if we alter the poller's advice by selecting connection  $n$  to transmit, they both contribute to  $\Delta Z_t$ . Our task is therefore to find an  $n$  (possibly  $m$ ) which minimizes the overall effect,  $(\hat{Q}_t^m - \hat{Q}_{-1,t}^m) - (\hat{Q}_t^n - \hat{Q}_{-1,t}^n)$ , or equivalently, to find an  $n$  which minimizes  $(\hat{Q}_t^n - \hat{Q}_{-1,t}^n)$ . It is noteworthy that such a minimum value is at most zero since  $m = n$ .

By doing so,  $\Delta Z_t$  is approximated by the overall effect. However, at two timepoints,  $t$  and  $t+1$ , the connections being evaluated are usually different in the original polling sequence. Therefore, the impact of action at  $t$  is usually not on the transition cost and state at  $t+1$  but is delayed to the next evaluation times for the connections  $m$  and  $n$ . In the Appendix, we show the rationale behind our  $\Delta Z_t$  approximation for handling such delayed impact problems.

## 6 SIMULATION RESULTS

The simulations are conducted with ns2 [20]. All the packets of video streams are delivered only when the access point/base station is in PCF mode, where the corresponding software module for ns2 is contributed by [1]. The bandwidth, beacon period, and maximum contention free period (CFP) duration are set as 54 Mbps, 10 ms, and 8 ms, respectively. At the start of a beacon period, the CFP begins. When there is nothing to send or the CFP duration exceeds the maximum CFP duration, the CFP is terminated and the base station enters DCF mode. The radio propagation model in ns2 is set as shadowing. The path loss exponent and shadowing deviation are set as 4 and 7, respectively. In addition,  $d_0$  is set to 10 and the communication radius is 500 m.

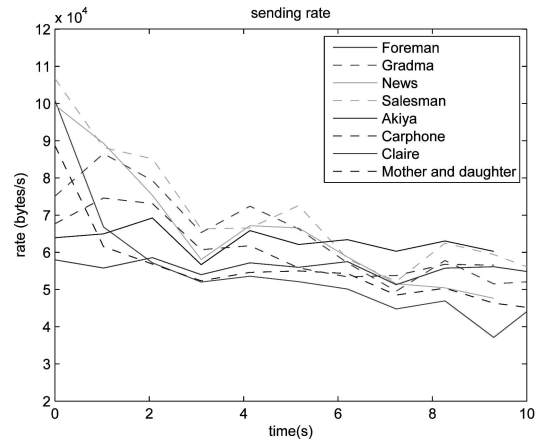


Fig. 2. The momentary sending rates of the encoded video streams.

The packets of the same video frame have a common id. TAEC checks the id of each packet and then distinguishes which frame it belongs to. We implement TAEC by modifying the priority queue provided by ns2 so that each connection is associated with the pushout-style queue dedicated to it. The buffer size of each queue is set as three video frames. Upon reception of an acknowledgment from the MAC layer, we update the channel condition as a good state. Moreover, the expiration of the sending timer in the MAC layer indicates a failure of transmission over the wireless channel, and thus we update the channel condition as a bad state. We simulate the video stream with GoP as IPPP. The I-frame (intra-coded frame) is encoded independently, but decoding the rest of the P-frames (predictive-coded frame) depends on the I frame in the same GoP. In deriving the policy, we arbitrarily let  $\nu = 8$  for the I frame and  $\nu = 2$  for the P frame. Furthermore, we let  $\hat{p}(g|b) = 0.1$ ,  $\hat{p}(b|g) = 0.2$ ,  $\hat{p}_{e,g} = 0$ , and  $\hat{p}_{e,b} = 1$ . We also configure  $\hat{p}_a(1) = 0.3$ . That is, we assume that each packet receives one chance to transmit, and thus there are  $(8 + 2 \times 3) * 30/4 = 105$  chances to deliver packets during a second. Therefore, between two consecutive chances, the frame arrival rate is approximately 0.3.

Eight classic video clips, *Foreman*, *Grandma*, *News*, *Salesman*, *Akiya*, *Carphone*, *Claire*, and *Mother and daughter* are included in our simulation. The video sampling rate is 30 fps with CIF resolution ( $176 \times 144$ ). The quality of received video is evaluated through the frameworks evalvid [11] and evalvid-ns2 [10]. Following the procedure in the tutorial for evalvid [11], we use ffmpeg to encode and decode the video streams. Moreover, we set the target bit rate as 256 Kbps (or  $3.2 \times 10^4$  bytes/s), but the bit rate of video streams output by ffmpeg (single pass) is quite a bit larger than the target bit rate. As instructed in the tutorial for evalvid-ns2 [10], the video frame is segmented into 1,000-byte payloads and then each payload is encapsulated with 28-byte header as a packet. Fig. 2 shows the byte rates of each encoded video stream.

There are two sets of connections, namely the normal connections and premium connections. Each set consists of eight connections, and each of the connections carries one distinct video clip among the selected eight video clips. The reserved bit rate of the premium connection is 1.5 times larger than that of the normal connection. The original schedule is generated by the *weighted fair queuing* (WFQ). Therefore, if borrowing is disabled, then, by WFQ, each

TABLE 1  
Average PSNR for Normal Connections

clip	TAEC	only WFQ	priority
Foreman	34.7	31.5	32.7
Grandma	40.5	36.0	40.4
News	37.3	31.0	37.6
Salesman	36.8	28.5	38.4
Akiya	45.0	36.5	44.5
Carphone	36.8	33.0	36.1
Claire	46.0	44.0	45.6
Mother and daughter	42.8	40.3	42.3

TABLE 2  
Average PSNR for Premium Connections

clip	TAEC	only WFQ	priority
Foreman	34.6	34.4	34.9
Grandma	40.6	40.6	40.6
News	39.7	37.6	40.2
Salesman	39.0	37.2	39.1
Akiya	45.0	45.0	44.9
Carphone	36.8	36.7	36.7
Claire	46.0	46.0	46.0
Mother and daughter	42.8	42.8	42.8

normal connection approximately receives 2-Mbps bandwidth and the premium connection receives 3-Mbps bandwidth. The video streams of all connections are generated by a common fixed node, forwarded by the same base station, and destined to the distinct mobile nodes, where the normal connections are destined to nodes with labels ranging from 0 to 7 and the premium connections are destined to those ranging from 8 to 15. From the connection with destination node 0 to the one with destination node 15, the fixed node initiates a connection to start to send the video stream every 2 ms. The fixed node directly links to the base station with 100 Mbps capacity and a propagation delay 2 ms. The base station is located at (100, 100), where the mobile node  $i$  is initially located at  $(130 - 4i, 70 + 4i)$ .

In our simulation, we assume that each connection requires the immediate bandwidth guarantee which is specified as follows: When WFQ selects a connection to transmit, the connection can lend its transmission chance to another only under the condition that its head-of-line is not a type "I" frame, and its last type "I" frame and its preceding frame have been successfully transmitted. That is to say, the connection is willing to share its bandwidth when the error propagation and consecutive frame loss is suppressed. Provided that the transmission chance is borrowed, the WFQ behaves as if the transmission is done by the borrowed connection. Thus, WFQ updates both the *finish time* of the borrowed connection corresponding to the size of the packet being transmitted and the reserved bit rate of the borrowed connection. The corrupted frame that occurs due to the loss of some packets is not decoded. In this case, the last completely decoded frame is then inserted as the corrupted frame.

We compare the TAEC with a technique without any control but with WFQ (denoted as "only WFQ") and the scheme of multiplication of priorities (denoted as "priority"). With only WFQ, pushout-style per-connection queue is still associated with each connection. However, the pushout operation is performed only when the overflow occurs. Furthermore, the borrowing is disabled. In the scheme of multiplication of priorities, two priorities are considered, namely the type priority and the channel priority. The type priority is 3 when the frame type of the head of line is "I" and 1 otherwise. The channel priority is 2 when the channel condition is good and 1 otherwise. The final priority is then the product of type priority and channel priority. The same transport architecture as in TAEC is associated and the scheduler does not select the connection without any frame to send. If borrowing is allowed, the scheduler then chooses the connection with the largest priority to transmit. When there is a tie, the scheduler favors the premium connection.

For the purpose of comparison, we calculate the PSNR between the luminance component  $Y$  of the decoded video frame of the video stream received in the receiver and that of the original video stream. As a result, the lower PSNR indicates the lower video quality.

Tables 1 and 2 display the simulation results collected during the first 10 seconds in terms of PSNR (dB).

As the results show, the worst average PSNR for TAEC is around 34.6 dB, which is considerably better than that of "only WFQ", which is 28.5 dB.

In the settings of our next simulation, we enable the mobile node to move randomly from its initial position and rerun the simulation 10 times. Prior to simulation, we compute the reference PSNR value sequence in the reconstruction of the encoded video as compared to the original raw video. The *PSNR loss* of a frame is then defined as the reference PSNR of the frame minus the PSNR which is computed for the frame reconstructed from the possibly corrupted video sequence received. In each run of the simulation, we collect the PSNR loss over the first 10 seconds. Due to the obvious performance deficiency of the scheme "only WFQ," our comparison is narrowed down to TAEC versus scheme "priority." Additionally, the average PSNRs for each premium connection under TAEC and "priority" are similar. Therefore, in Fig. 3, only the cumulative probability distributions (CDF) of the PSNR losses for normal connections are displayed. Each result, except for the one for the video stream "Salesman," shows that, in general, the probability of zero-quality-loss frames (i.e., PSNR loss is zero) is significantly larger for TAEC. The exception is mainly due to the excessive sending rate of the video stream "Salesman" in the starting period, as can be seen in Fig. 2. Such an effect shows that TAEC prefers not to give too much extra bandwidth to sustain the quality of a highly overloaded connection. The results demonstrate that TAEC succeeds in increasing the number of zero-quality-loss frames by exploring the multiuser diversity gain. Another point is that for each stream except "Salesman," the CDF of "priority" is bounded by that of TAEC when the value of CDF is less than 0.95. However, TAEC usually has higher PSNR loss with probability 0.05. This is because TAEC tends to give lower priority to the I frame than does "priority" if some P frames exist which could be restored with zero quality loss. Therefore, TAEC might result in more consecutive frame losses and thus degrade the video quality. Such a problem can be remedied by altering the cost definition and will be explored in a future study.

Finally, we show the delay distribution of the received video frames in each normal connection when TAEC is applied in the base station. We regard the delay of each

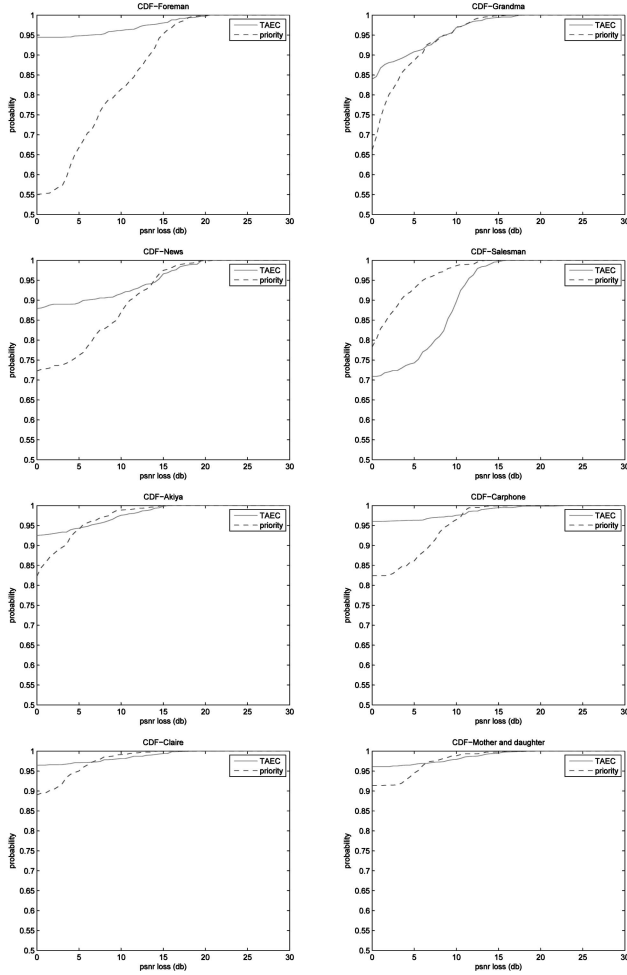


Fig. 3. The CDF of PSNR for video streams.

frame as the time the last bit of the frame is received in the mobile node minus the time the first bit of the frame is sent from the source application. As shown in Fig. 4, the delay for each normal connection is analogous to the uniform distribution over the range from 0.02 to 0.12 seconds.

## 7 CONCLUSION AND FUTURE WORK

We propose an error control method, TAEC, to support robust interactive video services for wireless channels. Specifically, TAEC consists of a transport architecture to satisfy the deadline constraint and a self-tuning controller to further improve performance. As the most salient feature, TAEC trades off and balances the expected future impact caused by the error control action under the current state in each connection. In our simulation, TAEC is embedded into an IEEE 802.11 PCF mode to perform the error control for downlink transmission. As demonstrated by the simulation results, while attaining the immediate bandwidth guarantee, TAEC also increases the number of received video frames which can be fully reconstructed. As a result, except for the connections which need much more bandwidth than granted to sustain their qualities, the video qualities of the other connections are improved in terms of average luminance PSNR.

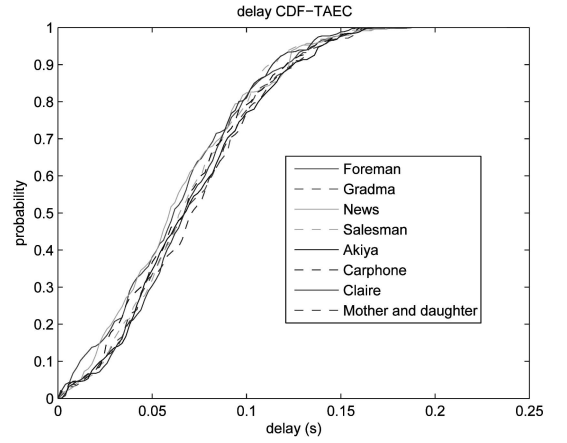


Fig. 4. The delay distribution of the received video frames in TAEC.

This study aims at reducing the overall cost, which is defined as the average number of distorted frames perceived by the users. One future research direction is the reformulation of TAEC to tackle other cost definitions, such as power consumption, and other video quality assessment metrics. Other possible future work could extend TAEC to take into account forward error correction, compressed real-time transport protocol (RTP), and encapsulating different frames into a single large packet.

## APPENDIX

To study the sample path property for the non-time-homogeneous Markov chain, first note that the transition matrices (or transition kernels if the state space is Borel) of the process under study are all given or can be derived in advance. However, they are time-dependent. By the Tulcea's Theorem [8], a unique probability measure exists over the sample paths and thus enables the study of such a property.

Given Markov chain  $X_t$  with time varying transition matrices, we define a difference process,  $Y_t$ , as follows:

$$\forall t \geq 1, \quad Y_t := \sum_{i=1}^{t-1} \left\{ c(X_i, X_{i+1}) - \eta_i \right\} + v_t(X_t),$$

where  $\eta_t$  is a scalar for every  $t > 0$  and  $v_t(\cdot)$  is some value function over state space. Let  $\mathcal{F}_t$  denote the smallest  $\sigma$ -algebra containing all the information up to time  $t$ . Process  $Z_t$  is then recursively defined as follows: For all  $t \geq 1$ ,

$$\begin{aligned} Z_{t+1} - Z_t &:= \mathbf{E}[Y_{t+1} - Y_t | \mathcal{F}_t] \\ &= \mathbf{E}[c_t(X_t, X_{t+1}) + v_{t+1}(X_{t+1}) | X_t] \\ &\quad - v_t(X_t) - \eta_t, \end{aligned}$$

and  $Z_1 := 0$ . Here, we generalize the notation  $c(\cdot, \cdot)$  to  $c_t(\cdot, \cdot)$  for considering the time-dependent cost function. We denote  $Z_t - Z_{t-1}$  by  $\Delta Z_t$ . Then,  $Z_t$  can be written as follows:

$$Z_t = \sum_{i=1}^{t-1} \Delta Z_i.$$

We then have the following lemma:

**Lemma 1 (upper bound).** *Suppose that there exists a finite number  $M$  such that  $|v_t(\cdot)| < M, 0 \leq c(\cdot, \cdot) < M$ . Then,*



$$\limsup_{t \rightarrow \infty} \frac{\sum_{i=1}^t c(X_i, X_{i+1})}{t} \leq \limsup_{t \rightarrow \infty} \frac{\sum_{i=1}^t \eta_i}{t} + \limsup_{t \rightarrow \infty} \frac{Z_t}{t}$$

almost surely.

This proof is very similar to the proof of Theorem 4.3 in [5] and thus we omit it.

Finding the optimal policy is of theoretical interest but is generally impractical. This being the case, we may resort to one of the following methods to find a near-optimal policy:

- **M1:** We stick to the bound  $\limsup_{t \rightarrow \infty} \frac{\sum_{i=1}^t \eta_i}{t}$  by letting  $\limsup_{t \rightarrow \infty} \frac{Z_t}{t} \leq 0$ .
- **M2:** When the bound  $\limsup_{t \rightarrow \infty} \frac{\sum_{i=1}^t \eta_i}{t}$  is not close to the optimal one, we attempt to reduce  $\Delta Z_t$  in an online fashion.

Let  $N$  denote the number of flexible communications and  $X_n(t)$  denote the state (coupled with the leftover budget information) of connection  $n$  at time  $t$ . In addition, let  $\underline{X}_t$  denote the overall system states  $(X_t^1, X_t^2, \dots, X_t^N)$ . Let  $\mathbb{X}^n$  denote the set where  $X_t^n$  takes the value and let  $\underline{\mathbb{X}}$  be defined as  $\prod_{1 \leq n \leq N} \mathbb{X}^n$ .

The goal of the Appendix is to show the rationale for deriving an improved policy when exploring multiuser diversity gain. That is, we show that, basically, what our procedure does is to apply **M2**. To do so, we first formulate process  $Y_t$ . From  $Y_t$ , we derive  $Z_t$ . We then give the reason why **M2** cannot be directly applied to minimize  $\Delta Z_t$ . Rather, we introduce another process,  $\hat{Z}_t$ , which deviates from  $Z_t$  within bounded distance. We then show how we approximate  $\Delta \hat{Z}_t$  and our decision is indeed to minimize the approximated  $\Delta \hat{Z}_t$ .

As follows, we consider the polling sequence which is altered by the borrowing action. Let  $\{\beta(i)\}_{i=1}^{\infty}$  be the original polling sequence and let  $\{\beta'_t(i)\}_{i=1}^{\infty}$  be the new polling sequence up to time  $t$  followed by the sequence  $\beta(t+1), \beta(t+2), \dots$ , and  $t$  is referred to as *current system time*. The idea behind the definition of  $\{\beta'_t(i)\}_{i=1}^{\infty}$  is prior to the system time. The precise new polling sequence can be obtained only after the system time. The new polling sequence cannot be known in advance and temporarily scheduled in the same manner as the original polling sequence. Moreover, if  $\beta(i) \neq \beta'_t(i)$ , we say the connection  $\beta'_t(i)$  borrows the transmission time from the connection  $\beta(i)$  at time  $i$ . In such a case,  $i \leq t$  is implied immediately and the error control action is always performed on connection  $\beta'_t(i)$  at time  $i$ .

Given the current system time,  $t$ , let *next evaluation times*  $\kappa_t(s)$  and  $\kappa'_t(s)$  after time  $s$  for connections  $\beta(s)$  and  $\beta'_t(s)$  be  $\inf\{\tau > s : \beta'_t(\tau) = \beta(s) \text{ or } \beta(\tau) = \beta(s)\}$  and  $\inf\{\tau > s : \beta'_t(\tau) = \beta'_t(s) \text{ or } \beta(\tau) = \beta'_t(s)\}$ , respectively. Note that for all  $t$ ,  $\kappa_t(s)$  and  $\kappa'_t(s)$  are finite by deadline-driven polling. In addition,  $\psi(s)$  and  $\psi'(s)$  are the *last evaluation times* prior to  $s$  for connections  $\beta(s)$  and  $\beta'_t(s)$ , respectively, i.e.,  $\sup\{\tau < s : \beta(s) = \beta(\tau)\}$  and  $\sup\{\tau < s : \beta'_t(s) = \beta'_t(\tau)\}$ , respectively.

Let the values  $c(X_s^n, X_\tau^n)$  and  $v(X_\tau^n)$  represent the cost and relative cost of connection  $n$  if its next evaluation time after  $s$  is  $\tau$ . Besides, denote  $\eta_\tau^n$  as the average cost,  $\eta$ , of solution to the Poisson equation for connection  $n$  at time  $\tau$ . Given the current system time,  $t$ , let  $C_t(i)$  denote the conditional expected system cost at time  $i$  as follows:

$$\begin{aligned} C_t(i) := & I(\beta(i) \neq \beta'_t(i)) \mathbf{E} \left[ c \left( X_{\psi'_t(i)}^{\beta(i)}, X_i^{\beta'_t(i)} \right) \right. \\ & + c \left( X_i^{\beta(i)}, X_{\kappa'_t(i)}^{\beta'_t(i)} \right) + c \left( X_i^{\beta(i)}, X_{\kappa_t(i)}^{\beta(i)} \right) \Big| \mathcal{F}_t \Big] \\ & + I(\beta(i) = \beta'_t(i)) \mathbf{E} \left[ c \left( X_i^{\beta(i)}, X_{\kappa_t(i)}^{\beta(i)} \right) \Big| \mathcal{F}_t \right]. \end{aligned}$$

By the property of conditional expectation, the immediate properties are listed as follows:

- $\mathbf{E}[C_t(i)|\mathcal{F}_s] = C_t(i)$  when  $s \geq t$ ;
- if  $\beta(i)$  is equal to  $\beta'_t(i)$  and  $\kappa_t(i) \leq t$ , then  $C_t(i) = c(X_i^{\beta(i)}, X_{\kappa_t(i)}^{\beta(i)})$ . If  $\beta(i)$  does not equal to  $\beta'_t(i)$ ,  $\kappa_t(i) \leq t$  and  $\kappa'_t(i) \leq t$ , then  $C_t(i) = c(X_i^{\beta(i)}, X_{\kappa'_t(i)}^{\beta'_t(i)}) + c(X_i^{\beta(i)}, X_{\kappa_t(i)}^{\beta(i)})$ .

Therefore, function  $C_t(i)$  represents the total transition costs of connections evaluated at time  $i$  which are either expected costs or realized costs, depending on whether the next evaluation times are after time  $t$  or not.

Define process  $Y_t$  as follows:

$$Y_t := v^{\beta(t)} \left( X_t^{\beta(t)} \right) + \sum_{i=1}^t C_t(i) - \eta_i^{\beta(i)}.$$

We investigate the following:

$$\begin{aligned} \mathbf{E}[Y_{t+1} - Y_t | \mathcal{F}_t] = & \mathbf{E} \left[ v^{\beta(t+1)} \left( X_{t+1}^{\beta(t+1)} \right) \right. \\ & + \sum_{i=1}^{t+1} C_{t+1}(i) - \eta_i^{\beta(i)} \Big| \mathcal{F}_t \Big] \\ & - \left( v^{\beta(t)} \left( X_t^{\beta(t)} \right) + \sum_{i=1}^t C_t(i) - \eta_i^{\beta(i)} \right). \end{aligned}$$

The values  $C_{t+1}(i)$  and  $C_t(i)$ , where  $i \leq t$ , are different only if the next evaluation time of connection  $\beta(i)$  and/or  $\beta'_{t+1}(i)$  is at time  $t+1$ , i.e., when  $i$  is  $\psi(t+1)$  and/or  $\psi'_t(t+1)$ . Fortunately, in considering  $\mathbf{E}[C_{t+1}(t+1)|\mathcal{F}_t] - C_t(t+1)$ , if  $\beta'_{t+1}(t+1) = \beta(t+1)$ , then the conditional expectation is zero. Therefore, we only take special consideration for the leftover terms in  $\mathbf{E}[C_{t+1}(t+1)|\mathcal{F}_t] - C_t(t+1)$  under the case  $\beta'_{t+1}(t+1) \neq \beta(t+1)$ . We then rewrite  $\mathbf{E}[Y_{t+1} - Y_t | \mathcal{F}_t]$  as follows:

$$\begin{aligned} Z_{t+1} - Z_t := & \mathbf{E}[Y_{t+1} - Y_t | \mathcal{F}_t] \\ = & \mathbf{E} \left[ c \left( X_{\psi(t+1)}^{\beta(t+1)}, X_{t+1}^{\beta(t+1)} \right) + v^{\beta(t+1)} \left( X_{t+1}^{\beta(t+1)} \right) \Big| \mathcal{F}_t \right] \\ & + I(\beta(t+1) \neq \beta'_{t+1}(t+1)) \\ & \left\{ \mathbf{E} \left[ c \left( X_{\psi'_{t+1}(t+1)}^{\beta'_{t+1}(t+1)}, X_{t+1}^{\beta'_{t+1}(t+1)} \right) + c \left( X_{t+1}^{\beta'_{t+1}(t+1)}, X_{\kappa'_{t+1}(t+1)}^{\beta'_{t+1}(t+1)} \right) \right. \right. \\ & \left. \left. - c \left( X_{\psi'_{t+1}(t+1)}^{\beta'_{t+1}(t+1)}, X_{\kappa'_t(t+1)}^{\beta'_t(t+1)} \right) \Big| \mathcal{F}_t \right] \right\} - v^{\beta(t)} \left( X_t^{\beta(t)} \right) - \eta_t^{\beta(t)}. \end{aligned} \quad (6)$$

By letting  $Z_0 = 0$ , process  $Z_{t+1}$  is defined recursively. It is easy to show that borrowing is not necessary if the following inequality holds:

$$\limsup_{t \rightarrow \infty} \frac{Z_t}{t} \leq 0, \quad (7)$$

then by Lemma 1, the limit sup of average cost can be shown to be bounded above almost surely by  $\lim_{t \rightarrow \infty} \sum_{i=1}^t \eta_{\beta(i)}/t$  which equals to  $\sum_{n=1}^N \eta_n / \phi_n$  by the property of deadline-driven polling. When applying **M2**, we attempt to reduce  $Z_t$

by minimizing  $\Delta Z_t$  and thus the average cost can be further reduced.

Note that the right-hand side of the last equality in (6) contains  $\beta(t+1)$  and  $\beta(t)$ . However,  $\beta(t+1)$  usually does not equal  $\beta(t)$ . Moreover, the next evaluation times cannot be known in advance at time  $t$  if borrowing is allowed. Instead, we consider another process,  $\hat{Z}_t$ , defined recursively as follows:

$$\begin{aligned} \hat{Z}_{t+1} - \hat{Z}_t &:= \mathbf{E} \left[ c \left( X_t^{\beta(t)}, X_{\kappa(t)}^{\beta(t)} \right) + v^{\beta(t)} \left( X_{\kappa(t)}^{\beta(t)} \right) \middle| \mathcal{F}_t \right] - v^{\beta(t)} \left( X_t^{\beta(t)} \right) \\ &\quad - \eta_t^{\beta(t)} + I(\beta(t) \neq \beta'_t(t)) \left\{ \mathbf{E} \left[ c \left( X_{\psi'_t(t)}^{\beta'_t(t)}, X_t^{\beta'_t(t)} \right) \right. \right. \\ &\quad \left. \left. + c \left( X_t^{\beta'_t(t)}, X_{\kappa'_t(t)}^{\beta'_t(t)} \right) \middle| \mathcal{F}_t \right] - \mathbf{E} \left[ c \left( X_{\psi'_t(t)}^{\beta'_t(t)}, X_{\kappa'_t(t)}^{\beta'_t(t)} \right) \middle| \mathcal{F}_{t-1} \right] \right. \\ &\quad \left. + \mathbf{E} \left[ v^{\beta'_t(t)} \left( X_{\kappa'_t(t)}^{\beta'_t(t)} \right) \middle| \mathcal{F}_t \right] - \mathbf{E} \left[ v^{\beta'_t(t)} \left( X_{\kappa'_t(t)}^{\beta'_t(t)} \right) \middle| \mathcal{F}_{t-1} \right] \right\} \end{aligned}$$

and  $\hat{Z}_0 := 0$ . The idea behind process  $\hat{Z}_t$  is that we first guess conditional expectation  $\mathbf{E}[v^{\beta(t)}(X_{\kappa(t)}^{\beta(t)})|\mathcal{F}_t]$  at time  $t$  in  $\hat{Z}_t$  and replace the value by a new one during each evaluation of the connection prior to  $\kappa(t)$ , i.e., borrowing transmission time from other connection before  $\kappa(t)$ . Instead of checking  $Z_t$ , we select  $\beta'_t(t)$  to minimize  $\hat{Z}_{t+1} - \hat{Z}_t$ . The following lemma rationalizes our approach:

**Lemma 2.** *There exists a constant  $M$  such that for any  $t > 0$ ,  $|\hat{Z}_t - Z_t| < M$ .*

**Proof.** Note that for each  $\hat{Z}_t$  and  $Z_t$ , the difference between them could only be one expected relative value, which is bounded, for each connection. Since there are  $N$  connections, the theorem follows.  $\square$

To correctly obtain  $\hat{Z}_{t+1} - \hat{Z}_t$ , we need to keep detailed information regarding the frame arrival and the time to  $\kappa(t)$ . To avoid this, we approximate it as follows: Denote  $\beta(t)$  by  $m$  and  $\beta'_t(t)$  by  $n$ . Consider the nontrivial case where  $\beta(t) \neq \beta'_t(t)$ . The term  $v^{\beta(t)}(X_t^{\beta(t)}) + \eta_t^{\beta(t)}$  indeed equals  $\hat{Q}_t^m$  due to Poisson's equation (2). The term  $\mathbf{E}[c(X_t^{\beta(t)}, X_{\kappa(t)}^{\beta(t)}) + v^{\beta(t)}(X_{\kappa(t)}^{\beta(t)})|\mathcal{F}_t]$  is approximated by  $\hat{Q}_{-1,t}^m$ . The term

$$\begin{aligned} \mathbf{E} \left[ c \left( X_{\psi'_t(t)}^{\beta'_t(t)}, X_t^{\beta'_t(t)} \right) + c \left( X_t^{\beta'_t(t)}, X_{\kappa'_t(t)}^{\beta'_t(t)} \right) + v^{\beta'_t(t)} \left( X_{\kappa'_t(t)}^{\beta'_t(t)} \right) \middle| \mathcal{F}_t \right] \\ - \mathbf{E} \left[ c \left( X_t^{\beta'_t(t)}, X_{\kappa'_t(t)}^{\beta'_t(t)} \right) + v^{\beta'_t(t)} \left( X_{\kappa'_t(t)}^{\beta'_t(t)} \right) \middle| \mathcal{F}_{t-1} \right] \end{aligned}$$

is approximated by  $\hat{Q}_t^n - \hat{Q}_{-1,t}^n$ .

## REFERENCES

- [1] A. Lindgren, Support for the PCF Mode of IEEE 802.11 for ns-2.1b8, <http://www.sm.luth.se/dugdale/index/software.shtml>, 2008.
- [2] S. Aramvith, C. Lin, S. Roy, and M. Sun, "Wireless Video Transport Using Conditional Retransmission and Low-Delay Interleaving," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 558-565, June 2002.
- [3] A. Bar-Noy, V. Drezin, and B. Patt-Shamir, "Efficient Periodic Scheduling by Trees," *Proc. IEEE INFOCOM*, 2002.
- [4] K. Chebrolu and R.R. Rao, "Selective Frame Discard for Interactive Video," *Proc. IEEE Int'l Conf. Comm. (ICC '04)*, June 2004.
- [5] X. Guo and P. Shi, "Limiting Average Criteria for Nonstationary Markov Decision Processes," *SIAM J. Optimization*, vol. 11, no. 4, pp. 1037-1053, Mar./May 2001.

- [6] R. Heddergott, U. Bernhard, and B. Fleury, "Stochastic Radio Channel Model for Advanced Indoor Mobile Communication Systems," *Proc. IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm. (PIMRC '97)*, pp. 140-144, Sept. 1997.
- [7] O. Hernandez-Lerma, J.I. Gonzalez-Trejo, and L.F. Hoyos-Reyes, "Minimax Control of Discrete-Time Stochastic Systems," *SIAM J. Control and Optimization*, pp. 1626-1659, 2003.
- [8] J. Hoffmann-Jorgensen, *Probability with a View toward Statistics*, vol. 2. Chapman & Hall, 1994.
- [9] R.A. Howard, *Dynamic Programming and Markov Processes*. MIT Press, 1960.
- [10] C.-H. Ke, C.-K. Shieh, W.-S. Hwang, and A. Ziviani, "An Evaluation Framework for More Realistic Simulations of MPEG Video Transmission," *J. Information Science and Eng.*, vol. 24, no. 2, pp. 425-440, 2008.
- [11] J. Klaue, B. Rathke, and A. Wolisz, "EvalVid—A Framework for Video Transmission and Quality Evaluation," *Proc. 13th Int'l Conf. Modelling Techniques and Tools for Computer Performance Evaluation*, pp. 255-272, <http://www.tkn.tu-berlin.de/research/evalvid/EvalVid/docevalvid.html>, Sept. 2003.
- [12] W.-K. Liao and Y.-H. Lai, "Development of Model-Based Control for Computer Communication Systems," technical report, 2009.
- [13] J.W.S. Liu, *Real-Time Systems*. Prentice Hall, 2000.
- [14] X. Liu, E.K.P. Chong, and N.B. Shroff, "Opportunistic Scheduling with QoS Requirements in Time-Varying Wireless Channels," *Proc. IEEE INFOCOM*, 2002.
- [15] I. Rhee, "Error Control Techniques for Interactive Low-Bit Rate Video Transmission over the Internet," *ACM SIGCOMM Computer Comm. Rev.*, vol. 28, no. 4, pp. 290-301, 1998.
- [16] H. Sanneck, M. Haardt, N. Le, and W. Mohr, "Selective Packet Prioritization for Wireless VoIP," *Proc. Fourth Int'l Symp. Wireless Personal Multimedia Comm.*, Sept. 2001.
- [17] R. Schwendener, "Indoor Radio Channel Model for Protocol Evaluation of Wireless Personal Area Networks," *Proc. IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm. (PIMRC '02)*, 2002.
- [18] J. Tang, L. Zhang, and C.-K. Siew, "An Opportunistic Scheduling Algorithm for MPEG Video over Shared Wireless Downlink," *Proc. IEEE Int'l Conf. Comm. (ICC '06)*, vol. 2, pp. 872-877, June 2006.
- [19] R.S. Typelly, J. Zhang, and E.K.P. Chong, "Opportunistic Scheduling for Streaming Video in Wireless Networks," *Proc. Conf. Information Sciences and Systems*, Mar. 2003.
- [20] VINT, The Network Simulator—ns-2, <http://www.isi.edu/nsnam/ns>, 2008.
- [21] H.-B. Yu, S. Yu, and C. Wang, "An Effective Architecture for Interactive Wireless Video Communication," *Proc. IEEE Vehicular Technology Conf. (VTC '04)*, vol. 5, pp. 2758-2762, May 2004.
- [22] S. Zachary, "Control of Stochastic Loss Networks, with Applications," *J. Royal Statistical Soc. Series B*, vol. 50, pp. 61-73, 1988.
- [23] Q. Zhang and S.A. Kassam, "Finite-State Markov Model for Rayleigh Fading Channels," *IEEE Trans. Comm.*, vol. 47, no. 11, pp. 1688-1692, Nov. 1999.



**Wei-Kuo Liao** received the BS degree in control engineering from the National Chiao Tung University, Taiwan, the MS degree in computer science from the National Tsing Hua University, Taiwan, in 1989 and 1993, respectively, and the PhD degree in computer engineering from Michigan State University in 1999. He is now a faculty member of the Department of Electrical Engineering, National Chiao Tung University, Taiwan. His main research interests include design of network architectures and model-based control for computer communication systems. He is a member of the IEEE.



**Yi-Hsuan Lai** received the PhD degree from the Institute of Statistics, National Chiao Tung University in 2004. She is a faculty member with Applied Mathematics Department at Hsuan Chuang University in Taiwan. Her research interests include the statistical compound decision and robust estimation.