Media-TCP: A Quality-Centric TCP-Friendly Congestion Control for Multimedia Transmission

Hsien-Po Shiang and Mihaela van der Schaar
Department of Electrical Engineering, UCLA, Los Angeles, USA
{hpshiang, mihaela}@ee.ucla.edu

Abstract—In this paper, we propose a quality-centric congestion control for multimedia streaming over IP networks, which we refer to as media-TCP. Unlike existing congestion control schemes that adapt a user's sending rate merely to the network condition, our solution adapts the sending rate to both the network condition and the application characteristics by explicitly considering the distortion impacts, delay deadlines, and interdependencies of different video packet classes. Hence, our media-aware solution is able to provide differential services for transmitting various packet classes and thereby, further improves the multimedia streaming quality. We model this problem using a Finite-Horizon Markov Decision Process (FHMDP) and determine the optimal congestion control policy that maximizes the long-term multimedia quality, while adhering to the horizon- $K$ TCP-friendliness constraint, which ensures long-term fairness with existing TCP applications. We show that the FHMDP problem can be decomposed into multiple optimal stopping problems, which admit a low-complexity threshold-based solution. Moreover, unlike existing congestion control approaches, which focus on maintaining throughput-based fairness among users, the proposed media-TCP aims to achieve quality-based fairness among multimedia users. We also derive sufficient conditions for multiple multimedia users to achieve quality-based fairness using media-TCP congestion control. Our simulation results show that the proposed media-TCP achieves more than 3dB improvement in terms of PSNR over the conventional TCP congestion control approaches, with the largest improvements observed for real-time streaming applications requiring stringent playback delays.

Keywords: Quality-centric congestion control, TCP-friendly congestion control for multimedia, finite-horizon Markov decision process, quality-based fairness.

I. INTRODUCTION

Transmission Control Protocol (TCP) is the most widely used protocol for data transmission at the transport layer. However, existing TCP congestion control provides dramatically varying throughput that is unsuitable for delay-sensitive, bandwidth-intense, and loss-tolerant multimedia applications [3] (e.g., real-time video streaming, videoconferencing etc.). This is due to the fact that current TCP congestion control aggressively increases the congestion window until congestion occurs, and then adopts an exponential backoff mechanism to mitigate the congestion. The fluctuating throughput results in long end-to-end delays which can easily violate the hard delay deadlines required by various multimedia applications. Hence, numerous multimedia transmission solutions over IP networks adopt User Datagram Protocol (UDP) at the transport layer [27]. However, UDP provides connectionless, unreliable services without guaranteed delivery, which limits the quality of service (QoS) support for multimedia applications at the transport layer. Therefore, multimedia applications need to rely on error resilience [8], forward error correction [9][10] and/or source coding rate control solutions [11][27], which need to be implemented at the application layer to achieve a desirable streaming quality. Moreover, the lack of congestion control mechanisms in UDP can lead to severe network congestion. Therefore, a significant body of existing multimedia streaming research over the past decade has focused on applying UDP-based congestion control that are TCP-friendly [4], which are being standardized as the Datagram Congestion Control Protocol (DCCP) [28]. However, these solutions often ignore the specific characteristics and requirements of multimedia applications, thereby leading to a sub-optimal performance for these applications.

Importantly, multimedia applications have several unique characteristics which need to be taken into account when designing a suitable congestion control mechanism. First, multimedia applications are loss-tolerant, and graceful quality degradation can be achieved if multimedia packets having lower distortion impacts are not received. Hence, various scheduling strategies [18][19] were proposed to optimize the received multimedia quality for multimedia streaming by prioritizing packets for transmission over error-prone IP networks. Such solutions, which explicitly consider multimedia packets’ distortion impacts, have also been adopted in order to improve the performance of congestion control mechanisms for multimedia applications [7]. Secondly, multimedia applications are delay-sensitive, i.e., multimedia packets have hard delay deadlines by which they must be decoded. If multimedia packets cannot be received at the destination before their delay deadlines, they should be purged from the senders’ transmission buffers to avoid wasting precious bandwidth resources. Third, in order to remove the temporal correlation existing in the source data, multimedia data are often encoded interdependently using prediction-based coding solutions (as in [20][21]). This introduces sophisticated dependencies between multimedia packets across time. Hence, if a multimedia packet is not received at the destination before its delay deadline, all the packets that depend on that packet should be purged from the transmission buffer to avoid unnecessary congestion, since these packets are not usable at the decoder side.

A. Limitations of current transport layer solutions for multimedia transmission

Supporting real-time multimedia transmission over IP networks is an important, yet challenging problem. Various approaches have been proposed to adapt the existing transport
layer protocols such that they can better support delay-
sensitive, loss-tolerant multimedia applications. However,
most current approaches still exhibit several key limitations.

1) Multimedia quality unaware adaptation. Conventional
transport layer congestion control control approaches are application-
agnostic, meaning that they merely attempt to avoid the
network congestion by adjusting the sending rates, without
considering the impact on the application’s performance. For
example, many TCP-friendly approaches apply analytical
models [2] on the long-term TCP throughput and adapt the
sending rate to the periodically updated TCP throughput
[3][4]. These model-based approaches aim to optimize the
bandwidth utilization, which may fail to maximize the
multimedia application performance (e.g. video quality) since
they do not consider multimedia characteristics, such as
distortion impact, delay deadline, etc.

2) Flow-based models for multimedia traffic without delay
consideration. Various approaches are proposed to adapt
multimedia applications to the available TCP throughput by
applying rate-distortion optimization [11], forward error
coding (FEC) [9][10], or frame dropping [25]. These solutions
often adopt flow-based models for multimedia traffic that only
consider the high-level flow rate (e.g. the average rate and
peak rate of the flow/frame [27]). They do not explicitly
consider the specific distortion impact and delay deadline of
each packet, as well as the interdependencies existing among
packets of multimedia applications. Hence, these congestion
control approaches only provide suboptimal solutions for
multimedia transmission [18][29].

3) Myopic adaptation. Prediction/estimation of the network
condition is widely used in congestion control mechanisms
based on network information feedback, e.g. in [4][11].
However, these solutions adapt the congestion window
myopically, i.e. based only on the current network condition.
Considering the packets’ delay deadlines and dependencies in
the transmission buffer, the congestion window size not only
impacts the immediate multimedia quality, but also impacts
the available packets in the buffer for future transmission.
Hence, it is important to consider not only the instantaneous
multimedia quality, but also how the immediate congestion
window size impacts the long-term expected quality in the
subsequent time slots. In [12], it was shown that the quality
can be improved by allowing temporary violation of the TCP-
friendliness, while later compensating the congestion control
to maintain long-term TCP-friendliness. They proposed a joint
source rate control and QoS-aware congestion control scheme.
However, this solution only considers multimedia source rates
and adopts a heuristic rate-compensation algorithm that cannot
optimally determine the required congestion window size.

In summary, a media-aware congestion control mechanism
that optimally determines the required congestion window size
to maximize the long-term multimedia quality in a look-ahead
(foresighted) rather than myopic manner is still missing.

B. Contribution of our solution and paper organization

In this paper, to overcome the abovementioned limitations,
the proposed media-TCP aims to make the following contributions:

1) Quality-centric packet-based congestion control. The
proposed media-TCP congestion control is quality-centric,
meaning that it aims specifically at maximizing the received
multimedia quality. Instead of applying a flow-based
multimedia model, our solution takes into account the
distortion impact and delay deadline of each packet, as well as
the packets’ interdependencies using a directed acyclic graph
(DAG) [17]. Importantly, instead of reactively adapting the
throughput, the proposed media-TCP actively and jointly
optimizes the congestion window size as well as the
transmission scheduling to provide differential services for
different packet classes. Performing this joint optimization is
very important in order to maximize the multimedia quality,
because the optimal congestion window size depends on the
transmission order of the multimedia packets in the
transmission buffer.

2) Foresighted adaptation using a Markov decision process
framework. We formulate the congestion control problem
using a Finite-Horizon Markov decision process (FHMDP)
framework in order to maximize the expected long-term
multimedia quality, under a long-term TCP-friendliness
constraint over the subsequent K time slots (i.e. horizon-K
TCP-friendliness). Such foresighted planning is essential for
multimedia streaming since it can consider, predict, and
exploit the dynamic characteristics of the multimedia traffic in
order to optimize the application performance over dynamic
IP networks. We show that the complex FHMDP formulation
can be decomposed into multiple optimal stopping problems
[23]. Based on the structural results obtained from the
decomposition, we present low-complexity threshold-based
algorithms when the multimedia packets are coded either
independently or interdependently.

3) Quality-based fairness among coexisting streams.
Preserving the fairness among the coexisting streams
represents an important issue [1][6]. However, even though
the throughput/bandwidth is equally shared by the users,
multimedia users can still experience very different qualities
since various applications and source data may result in
different traffic characteristics. Hence, instead of the
throughput-based fairness proposed by most existing
congestion control solutions, we focus in this paper on
quality-based fairness. We adopt Jain’s fairness index [26] on
the multimedia qualities and show that the proposed media-
TCP is able to achieve quality fairness among multimedia
users. In [25], the authors also proposed a frame dropping
scheme for min-max distortion fairness. However, the frame
dropping approach is determined myopically, without
considering the resulting TCP-friendliness to other flows.

In Table I, we compare the features of the proposed media-
TCP with the existing TCP-friendly congestion control
solutions for multimedia streaming.
The paper is organized as follows. In Section II, we first formulate the packet-based media-TCP congestion control problem for one media-TCP user. In Section III, we present the FHMDP framework used by the media-TCP user to determine the optimal transmission scheduling and congestion window size. In Section IV, we investigate how to decompose the FHMDP problem and provide structural results for solving this problem in different transmission scenarios. In Section V, we investigate multiple media-TCP users interacting in the same network with regular TCP users and discuss the quality-based fairness among the multimedia users. Simulation results are shown and discussed in Section VI, and Section VII concludes the paper.

II. MEDIA-TCP CONGESTION CONTROL PROBLEM FOR ONE MEDIA-TCP USER

A. Transport layer model

As in TCP, each packet transmission is acknowledged after a round-trip time (RTT) $Rtt$. Packet loss rate $p$ can be measured based on the packets’ acknowledgements. We assume a model-based congestion control as in [4][8][11] that adapts the congestion window size to a long-term available TCP throughput calculated from the packet loss rate $p$ and the round-trip time $Rtt$ (the adaptation will be discussed in Section II.D). Let $l$ represent the packet size. The long-term available TCP throughput can be approximated by

$$ R_{AIDM}(a, b, Rtt, p) = \frac{l(2 - b)}{Rtt \sqrt{2h p}} $$

(bits/sec) [14], where the TCP congestion control is modeled as a special case of generic Additive Increase Multiplicative Decrease (AIMD) based congestion control with parameters $(a, b)$ [14]. By substituting $(a, b)$ to $(1, 0.5)$, we have the well-known TCP response function $^2$ $R_{TCP}(Rtt, p) = \frac{l}{Rtt} \sqrt{\frac{3}{2p^2}}$ (bits/sec). We assume a time-slotted system and set the time slot duration $T$ as $Rtt^3$. Denote the measured packet loss rate in time slot $k$ as $p^k$. We define the expected TCP window size in time slot $k$ as $W_{TCP}^k(p^k) = R_{TCP}(Rtt, p^k) \frac{Rtt}{T} = \frac{3}{\sqrt{2p^k}}$ (pkts/time slot), which can be viewed as a metric describing the network congestion in time slot $k$.

B. Application layer multimedia model

Multimedia data is encoded and packetized into multiple data units at the application layer. A data unit usually encapsulates a video slice, which contains a set of macroblocks or an entire video frame (see e.g. the H.264 standard [21] for example). We assume that these data units will be packetized into RTP packets with the same size $l$ for transmission at the transport layer [4][14]. We also assume that these packets are classified into $M$ multimedia classes $\{CL_1, ..., CL_M\}$. These packets are queued in different transmission (post-encoding) buffers for transmission. A class $CL_m$ in time slot $k$ is characterized by the set of parameters $\psi_m = \{N_m^k, N_{mk}^L, N_{mk}^D, Q_m, depth_m^k\}$. These parameters are discussed next.

(a) Packet number: Let $N_m^k$ represent the number of packets in the transmission buffer of class $CL_m$ in time slot $k$.

(b) Arrival rate and discard rate: Let $N_{mk}^L$ denote the arrival rate, which represents the number of packets in class $CL_m$ that arrive in time slot $k$. Let $N_{mk}^D$ denote the discard rate, which represents the number of packets in class $CL_m$ whose delay deadline expires in time slot $k$. A packet is purged from the buffer if 1) it is successfully transmitted or 2) its delay deadline is expired. Based on the packet arrivals and departures, the number of packets

---

1. Given the current congestion window size $W$, the user increases its congestion window size as $W + \alpha$ per RTT when there is no packet loss. When a packet loss event occurs, the TCP congestion control decreases the congestion widow size as $(1 - b)W$ and retransmits the packets. A packet loss event can be either a timeout or receiving three duplicated acknowledgements in a row [14].

2. A more sophisticated TCP response function can be obtained by Markov chain modeling [2], which considers the timeout duration. For simplicity, we assume a fixed RTT (time slot) in this paper. However, the proposed approach can be easily extended in a time-varying RTT environment.
$N^k_m$ varies over time. In practice, if the multimedia data is pre-encoded, the arrival rate and discard rate in each time slot can be computed a priori. In the case of real-time multimedia transmission, e.g. video conferencing, the arrival time and the delay deadline can be stochastically modeled.

(c) Distortion impact: We assume an additive distortion reduction for the packets similar to the one employed in [17][18]. Let the distortion reduction when the packets in class $CL_m$ are received and decoded at the receiver be $N^k_m Q_m$ , where $Q_m$ represents the distortion impact of the class $CL_m$.

(d) Depth: Some classes of packets need to be received before others. Such interdependencies among the multimedia classes can be represented using a DAG [17]. Figure 1 gives an example of a DAG, in which MPEG video frames are classified into classes. More examples can be found in [17][18], and our solution is not restricted to any packet classification methods. Based on the DAG, if there is a path from class $CL_m$ to $CL_n$ , we say the class $CL_m$ is an ancestor of $CL_n$ , and $CL_n$ is a descendant of $CL_m$ . Denote $Anc_m$ and $Des_m$ as the ancestor set and descendent set of the class $CL_m$ . Let $depth^k_m$ represent the depth (the maximum distance) from class $CL_m$ to the root in the DAG in time slot $k$ . For classes at the root of the DAG, we define its depth to be 0. Depth captures the importance of a class in terms of interdependency, which depends on the depths of the ancestor classes, i.e. $depth_m = \max depth^k_m + 1$ .

The DAG structure varies over time as a traveling tree in [18]. Figure 1 also shows the variation of the DAG when the packets in class $CL_1$ and $CL_3$ are transmitted in time slot $k$.

\[ N^k_m = N^0_m I(kT \leq t_m \leq (k + 1)T), \]

where $N^0_m$ represents the initial size of the class. The discard rate can be written as $N^D_k = N^0_m I(kT \leq d_m \leq (k + 1)T)$.

Let $\pi^k_m \in \{1,0\}$ represent the transmission permission for transmitting the packets in class $CL_m$ in time slot $k$ . We assume that if $\pi^k_m = 1$, then all of the $N^k_m$ packets of class $CL_m$ are transmitted in time slot $k$ ; if $\pi^k_m = 0$, then no packets in $CL_m$ are transmitted.\(^5\) Denote $\tau^k_m(\pi^k_m) = N^k_m \pi^k_m$ as the source rate of class $CL_m$ in time slot $k$ , and denote $\tau^k = [\tau^k_m, m = 1,..., M]$ as the vector of rates for all the classes in time slot $k$ . In addition, let $\pi^k_{APP} = [\pi^k_m, i = 1,..., k, m = 1,..., M]$ represent the transmission permissions of all the classes from time slot 1 to time slot $k$ . Hence, the availability $\rho^k_m$ of the class $CL_m$ at the receiver in time slot $k$ can be computed by $\rho^k_m(\pi^k) = I(\sum_{i=1}^k \pi^i_m \geq 1)$, where $I()$ represents an indicator function. Based on the DAG, the actual distortion reduction for a class depends on whether or not its ancestors are available at the receiver. Hence, the actual distortion reduction of class $CL_m$ can be written as

\[ Q^m_{act}(\pi^k_{APP}) = Q_m \prod_{\pi^k_m \in \text{Anc}_m} \rho^k_m(\pi^k_{APP}) \]

and the resulting multimedia distortion reduction in time slot $k$ can be represented by:

\[ Q^k(\pi^k_{APP}) = \sum_{m=1}^M Q^m_{act}(\pi^k_{APP}) \rho^k_m(\pi^k_m). \]

C. Conventional flow-based solutions

Most existing TCP-friendly congestion control solutions for multimedia streaming reactively adopt the available TCP throughput as a rate-budget constraint and maximize the immediate multimedia quality at the application layer (e.g. the rate-distortion optimization in [11] and the packet size adaptation in [7]). These solutions can be formulated using the following flow-based optimization.

**Myopic-Flow-Based Optimization\(^6\):**

\[ \max_{\pi^k_m, m = 1,..., M} Q^k(\pi^k_{APP}) \]

s. t. \[ \sum_{m=1}^M \pi^k_m \leq R_{TCP}(Rtt, p^k) \].

Note that these solutions passively adapt the available TCP throughput $R_{TCP}(Rtt, p^k)$ to the network condition $p^k$ in each time slot. In contrast, we aim to propose a congestion control mechanism that adapts the congestion window size to both the network congestion and the specific characteristics

\(^5\) For illustration simplicity, we consider only binary transmission permission $\pi^k_m$ in this paper to transmit the entire class or not. However, similar approach can be applied to transmit partial data of a class by considering $\pi^k_m \in \{0,1\}$.

\(^6\) We assume that the information of actual distortion reduction $Q^m_{act}(\pi^k_{APP})$ is available in both equation (2) and (4). Moreover, because of the retransmission error control in the transport layer, we ignore the impact of packet loss in the distortion reduction model.
and requirements of multimedia applications. Instead of shaping the traffic at the application layer to match the available throughput $R_{TCP}(R_{tt}, p^k)$, the proposed media-TCP jointly optimizes the congestion window size $W^k$ and the transmission permissions $\pi^k_m$ of classes at the transport layer to maximize the expected multimedia quality. In the next subsection, we discuss the media-TCP congestion control problem in more details.

**D. Proposed packet-based media-TCP solution**

The proposed media-TCP congestion control is illustrated in Figure 2.

**At the application layer:** Multimedia RTP packets are classified into $M$ classes based on their interdependencies. Sophisticated packet classification can be performed in the application layer based on the different video coding transmission permissions relying on the following two components: selects the classes of packets to transmit and also determines control, media-TCP adjusts the congestion window size will drop all the expired packets in the transmission buffer. However, unlike TCP which control as TCP, which retransmits the lost packets based on negative acknowledgements. At the transport layer:

**Network estimator:** The network estimator updates the network congestion metrics in each time slot as in TCP. For example, a simple updating rule of the packet loss rate can be written as $[11]: p^{k+1} = p^k + (1 - \alpha)\hat{p}^k(W^k)$, where $\hat{p}^k(W^k)$ represents the realization of the packet loss rate in time slot $k$, and $\alpha$ represents the updating rates. In this paper, we focus on the joint optimization of the congestion window size and transmission scheduling by the transmission scheduler. For exposition simplicity, we assume that the only network congestion metric is the expected TCP window size $W^k_{TCP}(p^k)$. Note that other congestion metrics and more sophisticated updating rules (as in [4][11]) can easily be integrated into the proposed media-TCP.

![Fig. 2 System diagram of media-TCP in time slot $k$ and $k + 1$.](image)

We assume that the proposed media-TCP congestion control adheres to the following TCP-friendliness constraint.

**Definition 1 Horizon-$K$ TCP-friendliness:** A congestion control scheme is horizon-$K$ TCP-friendly, if and only if the congestion control window sizes from time slot $i$ to time slot $i + K - 1$ satisfy the condition

$$\sum_{k=i}^{i+K-1} \frac{W_k}{W^k_{TCP}(p^k)} \leq K.$$

The horizon-$K$ TCP-friendliness constraint keeps the average TCP-friendliness ratio $W^k/W^k_{TCP}$ close to 1 over the horizon $K$. Based on this definition, in time slot $i$, the proposed media-TCP congestion control solves the following packet-based optimization.

**Foresighted-Packet-Based Optimization:**

$$\max_{x^k \in \mathbb{R}^M} \sum_{m=1}^{M} E[Q^k(\pi^k_m(x^k), m = 1, \ldots, M)]$$

subject to

$$\frac{1}{K} \sum_{k=i}^{i+K-1} E\left[\frac{W_k(x^k)}{W^k_{TCP}(p^k)}\right] \leq 1, \quad 0 \leq W_k(x^k) \leq W_{max}$$

where $E[\cdot]$ represents the expected value and $W_{max}$ represents the maximum congestion window size.

Comparing our media-TCP using the foresighted-packet-based optimization in equation (4) with the conventional solutions using the myopic-flow-based optimization in equation (2), the differences are:

1) Conventional solutions passively adapt the congestion

---

7 A positive priority metric for a class indicates that the benefit (i.e. the resulting expected distortion reduction) of transmitting the packets in that class is greater than the cost of transmitting the packets. Hence, transmitting the packets of that class increases the total utility.

8 Here, we assume that $R_{tt}$ remains constant. In practice, the round-trip time can also be updated using a similar updating rule. Our Theorems and Lemmas still hold in such cases.
window size to the network condition (e.g. the expected packet loss rate $p^k$). Our proposed media-TCP solution takes one step further by adapting the congestion window size to the network conditions as well as to the application characteristics by optimizing the transmission scheduling. This allows the user to adapt the congestion window size to the characteristics of the multimedia packets in the buffer to provide differential services for various packet classes. Hence, a packet class with a higher chance to be transmitted, which is desirable for maximizing the received multimedia quality.

2) Media-TCP maximizes the expected long-term distortion reduction over a horizon-$K$ instead of solely maximizing the immediate distortion reduction as in the conventional solutions. This is especially important for multimedia applications with content dependencies. For example, in the IBBP frame structure in Figure 1, the media-TCP user may want to plan the congestion window sizes for transmitting I and P frames, instead of myopically determining window sizes for the B frames.

3) Instead of performing a constrained rate optimization myopically at every time slot, media-TCP adopts the horizon-$K$ TCP-friendliness constraint. Note that the horizon-$K$ TCP-friendliness becomes the traditional rate budget constraint in equation (2) when $K = 1$. The larger horizons provide long-term TCP-friendliness, which leads to more flexible window sizes and a better expected long-term quality. However, the short-term TCP-unfriendliness can be high (especially when network condition is good, i.e. $W^k_{TCP}$ is large) and needs to be compensated in the subsequent time slots [12].

In the next section, we discuss how the foresighted-packet-based optimization in equation (4) can be solved by applying an FHMDP framework.

III. FINITE-HORIZON MARKOV DECISION PROCESS

In this section, we first formulate the media-TCP congestion control problem in equation (4) using an FHMDP with Markovian state transition. The complexity of the FHMDP can be high due to the large state space. Hence, in the next section, we will decompose the problem into simpler sub-problems having smaller state spaces. We model this problem as an FHMDP due to the following reasons:

1) In numerous multimedia applications, multimedia traffic can be described by Markov models (as in [24]).

2) TCP operations are commonly modeled by discrete-time finite-state Markov chains (see e.g. [2][5][15]). Hence, the average TCP window size can be described by Markovian models based on the states of all the users (or the aggregate states as in [9]) in the network.

The FHMDP framework can be defined by the tuple $\{\mathcal{A}, \mathcal{S}, P, u, \gamma, \lambda, K\}$ . The various components of the framework are described next:

(a) Action: We denote the action of the FHMDP in time slot $k$ as $a^k = \{\pi^k_m, m = 1, \ldots, M\} \in \mathcal{A} = \{0,1\}^M$.

(b) State and state transition: We denote the state of the FHMDP in time slot $k$ as $s^k = \{W^k_{TCP}, N^k\} \in \mathcal{S}^{Net} \times \mathcal{S}^{App} = \mathcal{S}$ , where the expected TCP window size $W^k_{TCP}$ represents the network state and the number of packets $N^k = \{N^k_m, m = 1, \ldots, M\}$ in all the packet classes represents the application state. Let $\mathcal{S}^{Net} = \{0,\ldots,W^{max}\}$ represent the state space of the network state $W^k_{TCP}$ , and let $\mathcal{S}^{App} = \{0,\ldots,N^{max}\}^M$ represent the state space of the application state, where $W^{max}$ represents the maximum number of the window size and $N^{max}$ represents the maximum number of packets in a class. Let $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0,1]$ denote the state transition function, which can be described for the network state and application state as follows:

1) The network state transition is described by the state transition probabilities $P(W^{k+1}_{TCP} | W^k_{TCP})$ , which can be evaluated by estimating the next possible packet loss rate $p^{k+1}$ given the current feedback $p^k$ as in [9][15]. In general, the number of TCP users in the network is large enough such that the network state transition is not impacted by a single user’s action.

2) The application state transition is described by the state transition probabilities $P(N^{k+1} | N^k, a^k)$ . The number of packets in each class varies over time depending on the action $a^k$ . Note that each class can have its own arrival rate per time slot $N^k_{m,A} \geq 0$ and its own discard rate per time slot $N^k_{m,D} \geq 0$ . Therefore, the application state transition can be computed as:

$$N^k_{m,A} = N^k_{m} (1 - \pi^k_{m}) + N^k_{m,A} - N^k_{m,D} (1 - \pi^k_{m})$$

$$= \frac{(N^k_{m} - N^k_{m,D})(1 - \pi^k_{m}) + \sum N^k_{m,A}}{\text{remaining number of packets}}$$

This framework can be applied to both pre-encoded and real-time multimedia applications. For pre-encoded multimedia applications, the media-TCP user knows the state transitions for the entire multimedia session and solves the finite-horizon dynamic programming problem. For real-time multimedia applications, media-TCP can apply stochastic models to capture the state transitions of the applications [18].

Since the network state transition and the application state transition are independent, we define the overall
state transition probabilities $^9$ as:

$$P(s^{k+1} \mid s^k, a^k) = P(W_{TCP}^{k+1} \mid W_{TCP}^k)P(N_k^{k+1} \mid N_k^k, a^k).$$

(c) **Utility, discount factor, and horizon definitions:**

First, we apply a positive Lagrangian multiplier $\lambda$ and modify equation (4) into an unconstrained optimization$^{10}$:

$$\max_{a^k \in A} \sum_{k=1}^{i+K-1} \mathcal{Q}^k(a^k, s^k) - \lambda \left( \sum_{k=1}^{i+K-1} \frac{W_k}{W_{TCP}^k} - \sum_{k=1}^{i+K-1} 1 \right),$$

(6)

where $u^k(a^k, s^k) = \mathcal{Q}^k(a^k, s^k) - \lambda \left( \frac{W_k}{W_{TCP}^k} - 1 \right)$ is referred to as the instantaneous utility in time slot $k$. The second term of the instantaneous utility can be interpreted as the TCP window size deviation cost. The Lagrangian multiplier $\lambda$ $^{11}$ determines how the media-TCP user favors the TCP-friendliness over the multimedia quality. Based on the unconstrained optimization, the objective of the FHMDP in time slot $i$ is defined as:

$$\chi^i(s^i) = \max_{a^i \in A} \sum_{k=i}^{i+K-1} \gamma^{k-i}u^k(s^k, a^k),$$

(7)

where $\chi^i(s^i)$ represents the optimal congestion control policy given the state $s^i$ at the time slot $i$, and $\gamma$ represents the discount factor ($0 \leq \gamma \leq 1$). Note that equation (7) is equivalent to the unconstrained optimization in equation (6) when $\gamma = 1$. Since the Markovian models of the network state and application state transition may not be accurate, the discount factor $\gamma$ is set smaller than 1 to alleviate the impact of the inaccurate future utilities. The tradeoff between the TCP-friendliness and multimedia quality with different $\lambda$ and $\gamma$ will be discussed in Section VI.

Note that the media-TCP not only maximizes the instantaneous utility, but also the expected future utilities, which are expressed using the expected utility-to-go, which is defined next.

**Definition 2. Expected utility-to-go:** Define the expected utility-to-go at the last time slot of the horizon as

$$J^i_\mu(s^i) = u^{i+K-1}(s^i, \mu(s)), \forall s^i \in \mathcal{S},$$

where $\mu : \mathcal{S} \rightarrow \mathcal{A}$ represents a stationary mapping from the given state to an action. We define the expected utility-to-go in time slot $k = i, \ldots, i + K - 2$ as:

$$J^i_\mu(s^k) = u^k(s^k, \mu(s^k)) + \gamma \sum_{s^{k+1} \in \mathcal{S}} P(s^{k+1} \mid s^k, \mu(s^k))J_{\mu}^{k+1}(s^{k+1}).$$

The optimal congestion control policy in time slot $i$ can then be rewritten as:

$$\chi^i(s^i) = \arg \max_\mu J^i_\mu(s^i).$$

(8)

The system diagram of the proposed media-TCP congestion control using FHMDP framework is shown in Figure 3. The media-TCP user repeats the following steps at each time slot $i$:

1) Calculate the application state and network state transition probabilities.
2) Evaluate the expected utility-to-go at the various time slots of the entire horizon, i.e. from $J^i_\mu(s^i) \in \mathcal{S}$ to $J^{i+1}_\mu(s^{i+1}) \in \mathcal{S}$.
3) Based on the expected utility-to-go, the user updates the policy using

$$\mu(s) = \arg \max_\mu \left\{ u(s^i, a) + \gamma \sum_{s^{i+1} \in \mathcal{S}} P(s^{i+1} \mid s^i, a)J_{\mu}^{i+1}(s^{i+1}) \right\}.$$ (9)

and obtains the optimal action given the current state $s^i$.

The approach is similar to the “receding horizon control” in the optimal control literature [15]. Note that the complexity of solving the FHMDP directly is extremely high: it is proportional to the square of the number of states, and the number of states is exponential in the number of classes, i.e. $(N_{\max})^M$. Hence, it is important to decompose the FHMDP problem into sub-problems with smaller state space to reduce the complexity of solving the problem.

---

$^9$ In this work, we assume that the state transition probabilities can be calculated a priori. In fact, the state transition probabilities can be learned on the fly as long as the transitions of the states are semi-stationary.

$^{10}$ To simplify notation, we ignore the expectation $E[\mu]$ in our FHMDP formulation.

$^{11}$ $\lambda$ can be chosen based on the dual problem of equation (4) with the constraint $\sum_{m=1}^{M} \left( Q_m - \lambda W_{TCP}^k \right) N_m^k = 0$, which suggests $\lambda = \frac{W_{TCP} \tilde{Q}_m}{m}$, where $\tilde{Q}_m$ represents the average distortion reduction per packets in class $CL_m$, and $W_{TCP}$ is the time average $W_{TCP}^k$ over the horizon.
congestion control problem described in Section II.D to facilitate a low-complexity threshold-based congestion control as shown in equation (3). In Section IV.A, we discuss the decomposition of the FHMDP when the applications are coded independently. Then, in Section IV.B, we take the interdependencies among packets into account and discuss how to derive the structural results in this case.

A. Decomposition with independently coded packets

We first discuss the case when all the packets are independently coded (for example, video streams are coded using motion-JPEG), i.e. all the classes have the same depth, $depth_m = 0, m = 1, ..., M$. We first examine the structure of $J^k_{\mu,m}(s^k)$ and define the following property of the expected utility-to-go.

**Definition 3. Separable expected utility-to-go:** The expected utility-to-go $J^k_{\mu,m}(s^k)$ in time slot $k$ is separable if and only if it can be written in the form of

$$J^k_{\mu,m}(s^k) = \sum_{m=1}^{M} J^k_{\mu,m}(W^k_{TCP}, N^k_m) + C(W^k_{TCP}),$$

where $J^k_{\mu,m}(W^k_{TCP}, N^k_m)$ represents the utility-to-go component of a specific class $CL_m$, and $C(W^k_{TCP})$ represents a term that only depends on the network state $W^k_{TCP}$.

Next, we show that the expected utility-to-go is separable for multimedia applications with independently-coded packets.

**Theorem 1:** The expected utility-to-go $J^k_{\mu,m}(s^k), \forall s^k \in S, k = i, ..., i + K - 1$ over the horizon are separable, and the utility-to-go component $J^k_{\mu,m}(W^k_{TCP}, N^k_m)$ of class $CL_m$ is a nondecreasing function of $N^k_m$.

**Proof:** See Appendix B.

Based on Theorem 1, we have the following remarks:

**Remark 1:** The separation of the expected utility-to-go $J^k_{\mu,m}(s^k)$ in Theorem 1 suggests that the utility-to-go evaluation can be decomposed into $M$ independent FHMDP sub-problems. Each sub-problem computes a utility-to-go component $J^k_{\mu,m}(W^k_{TCP}, N^k_m)$ for a class (see equation (17) for the optimization sub-problems). The decomposition significantly reduces the overall complexity originally proportional to $(N^{max})^M$ to a complexity proportional to $MN^{max}$.

**Remark 2:** The nondecreasing property of $J^k_{\mu,m}(W^k_{TCP}, N^k_m)$ in $N^k_m$ in Theorem 1 shows that the more packets in the transmission buffer of class $CL_m$, the media-TCP user has higher expected utility for the specific class when the application is independently coded. Next, we investigate how the user determines the optimal congestion control policy when it knows the number of packets in the transmission buffer of each class. The following theorem presents the structural results of solving the FHMDP problem.

**Theorem 2. Structural results of the proposed media-TCP with independent packets:** Given the state $s^i$ in time slot $i$, the optimal policy of the transmission permission for a class $CL_m$ is $\pi^*_m(s^i) = I(PM^*_{m}(s^i) > 0)$, where the optimal priority metric $PM^*_{m}(s^i)$ is computed by:

$$PM^*_{m}(s^i) = \left\{ Q_m - \lambda \frac{\gamma}{W^k_{TCP}} \right\} N^k_m + \sum_{s^i \in S} \gamma P(W^k_{TCP} | W^k_{TCP}) J^k_{\mu,m}(W^k_{TCP}, N^k_m) - J^k_{\mu,m}(W^k_{TCP}, N^k_m - (N^k_m - N^{k^A}_m)) \right\}$$

**Proof:** See Appendix C.

Theorem 2 indicates that when the packets are coded independently, the media-TCP congestion control problem becomes an optimal stopping problem [23], where the media-TCP user transmits the packets of a certain class if and only if the priority metric of the class is positive. The priority metrics $\chi^*(s^i) = [PM^*_{m}(s^i), m = 1, ..., M]$ quantify the benefits of transmitting packets from various classes as opposed to not transmitting them in time slot $i$. Importantly, in addition to the distortion impact $Q_m$, the media-TCP user needs to consider the arrival rates and discard rates of the various classes. If a class $CL_m$ has numerous expiring packets (i.e., $N^k_m - N^{k^A}_m$ is large), it can be shown that the respective class has a larger $PM^*_{m}(s^i)$ to be transmitted in time slot $i$, instead of waiting for a future time slot. Moreover, it can be shown that at a time slot with a better network state (larger $W^k_{TCP}$), we have larger priority metrics from equation (10) for all the classes and hence, more classes are able to obtain the transmission permissions. Based on Theorem 2, we have the following remarks:

**Remark 3:** In equation (10), as $\gamma$ approaches 0, the user prefers to prioritize the packet classes based on their distortion impact values $Q_m$. As $\gamma$ approaches 1, the user increasingly weights the impact from the arrival rate and discard rate on the future expected utility. Note that when $\gamma = 0$, the FHMDP problem in equation (7) becomes a myopic optimization that merely optimize the instantaneous utility, which is equivalent to solving an unconstrained optimization of the conventional solution in equation (2).

**Remark 4:** Note that the optimal congestion control policy $\chi^*(s^i)$ in equation (7) includes the optimal priority metrics $\chi^*(s^i)$ and the congestion window size $W^*(s^i)$. Theorem 2 provides the optimal priority metrics $\chi^*(s^i) = [PM^*_{m}(s^i), m = 1, ..., M]$. Based on this, the optimal congestion window size $W^*(s^i)$ of the media-TCP user can be written as $W^*(s^i) = \sum_{m=1}^{M} N^k_m I(PM^*_{m}(s^i) > 0)$, and the resulting
expected multimedia quality can be computed by
\[ \bar{Q}(s') = \sum_{m=1}^{M} Q_m N_m I(\rho M_m^*(s') > 0) . \]
Note that the optimal policy varies with both the application state and the network state in time slot \( i \) \( (s' = \{W_{TCP}^i, N_i^i\}) \).

In Appendix A, Algorithm 1 provides the specific procedures for computing the optimal congestion policy \( \pi^*(s') \) when the packets are independent. The time complexity of the algorithm is \( O(MKN_{\text{max}}^2) \).

B. Decomposition with interdependently coded packets

In this subsection, we investigate the decomposition of FHMDP problem when the packets have interdependencies, described by a DAG as introduced in Section II.A. The following theorem presents the structural results of solving the FHMDP problem.

**Theorem 3. Structural results of media-TCP with interdependent packets:** Given the DAG and the state \( s' \) in time slot \( i \), the FHMDP problem can be solved by repeating the following two phases:

**Phase 1.** Select packet classes to transmit at the current time slot at the depth \( \text{depth}_m^i = j - 1 \):
\[ \pi_m^i(s') = I(\rho M_m^i(s') > 0), \forall CL_m \in \{CL_m, \text{depth}_m^i = j - 1\} , \]
where
\[ PM_m^i(s') = \left( \frac{Q_m}{W_{TCP}^i} \right) N_m^i + \gamma \sum_{s' \in S} P(W_{TCP}^{i+1} | W_{TCP}^i) \left( \frac{j_{m,i}^{i+1}(W_{TCP}^{i+1}, N_m^i) - j_{m,i}^{i+1}(W_{TCP}^{i+1}, N_m^i - (N_{D,i} - N_{A,i}))}{j_{m,i}^{i+1}(W_{TCP}^{i+1}, N_m^i)} \right) . \]

and \( j \) represents the number of iterations.

**Phase 2.** Update the actual distortion impact of each class:
\[ Q_m^{act} = Q_m \prod_{\forall CL_m \in \text{Desc}_m} \pi_m^i, m = 1, ..., M . \]

**Proof:** The proof can be easily provided by considering the DAG structure. We omit the proof due to space limitations.

In Phase 1, the media-TCP user selects packet classes for transmission by applying Theorem 3 starting from the classes at the root of the DAG. Since classes with the same depth are independent of each other, Theorem 2 can be applied to Phase 1 for classes with the same depth. Phase 2 indicates that if a class has no transmission permission, i.e. \( \pi_m^i(s') = 0 \), the media-TCP user set all its descendents’ distortion impact to 0, i.e. \( Q_m^{act} = 0 \) for \( \forall CL_m \in \text{Desc}_m \) (see Section II.B). Based on the DAG, since the distortion impact of a class is only influenced by the ancestors, the greedy algorithm in Theorem 3 starting from the root provides the optimal congestion policy. The two phases are repeated until the maximum depth of the DAG is reached.

In Appendix A, Algorithm 2 provides the procedures for computing the optimal congestion control policy \( \chi^*(s') = \{x^*(s'), W^*(s')\} \) when the packets are interdependently coded. Assuming that the DAG has the maximum depth \( \bar{D} \) and the number of classes per depth is \( \bar{M} \) on average, the complexity of the algorithm can be represented by \( O(MD\bar{K}(W_{\text{max}}N_{\text{max}}^2)) \).

V. FAIRNESS AMONG MULTIPLE MEDIA-TCP STREAMS

In the previous sections, we focus on only one media-TCP user interacting with multiple regular TCP users in the same network. In this section, we assume that there are \( V \) media-TCP users interacting with other TCP users in a network and investigate the competition among the multimedia users. Denote \( V = \{V_n, n = 1, ..., V\} \) as the set of the media-TCP users. Denote user \( V_n \) ’s expected multimedia distortion reduction in time slot \( k \) as \( \bar{Q}_n^k \). We assume a saturated condition (i.e. all the users continuously have their source traffic fed into their transmission buffers). We apply the well-known Jain’s fairness index [26] to quantify the fairness among the \( V \) media-TCP users:
\[ \mathcal{F}^k = \frac{\sum_{n=1}^{V} Q_n^k}{\sum_{n=1}^{V} (Q_n^k)^2} . \]

The fairness index \( \mathcal{F}^k \) measures the quality deviation of the multimedia applications. It varies as the media-TCP users make their own decisions at each time slot. Note that the index is always bounded by 1. The quality-based fairness is reached, i.e. \( \mathcal{F} = 1 \), only when all the media-TCP users have the same multimedia quality.

Following the TCP response function introduced in Section II.B and the packet loss rate updating rules in Section II.D, the expected network state of a user \( V_n \) in the next time slot can be expressed as \( E[W_{TCP}^{k+1}(p_k, W_n^k)] = \frac{1.5}{p_k^{k+1}(p_k, W_n^k)} \).

Similarly, we denote \( E[PM_{mn}^{k+1}] \) as the expected priority metric of \( CL_{mn} \) (the \( m \)-th class of user \( V_n \) ) in time slot \( k + 1 \), which can be shown as a function of \( W_n^k \). Then, we can prove the following lemma.

**Lemma 1:** For user \( V_n \), its priority metrics \( \{E[PM_{mn}^{k+1}(W_n^k)], \forall CL_{mn} \in V_n\} \) are all nonincreasing functions of \( W_n^k \), if
1) \( \gamma = 0 \), or
2) \( 0 < \gamma \leq 1, N_{D,m}^k = N_{A,m}^k, \forall CL_{mn} \in V_n, k = 1, ..., K \)

**Proof:** For both conditions, the priority metric of class \( CL_{mn} \) can be written by \( E[PM_{mn}^{k+1}(W_n^k)] = Q_{mn} - \lambda E[W_{TCP}^{k+1}(p_k, W_n^k)] \) based on equation (10). Since the estimated packet loss rate \( p^{k+1}(p_k, W_n^k) \) is in general a monotonically nondecreasing function of the congestion window size \( W_n \) [11], it is straightforward that both the

\[\text{12} \] The packet loss rate can be modeled as an M/M/1/K queue at the bottleneck link that reacts to the summation of the window sizes of all the users.
expected window size \( E[W_{n+1}^k] \) and the expected priority metrics \( E[P_{mn}^{k+1}(W_n^k)] \) in the next time slot are nonincreasing functions of \( W_n^k \) when the users apply the myopic media-TCP or when the packets in the buffer expire in the next time slot. In these two cases, the priority metrics are dominated by the first term in equation (10). The second condition requires the delay deadlines of the classes to be stringent, which is more likely to be true in the case of real-time streaming, as opposed to the pre-encoded streaming applications. In these two cases, Lemma 1 indicates that the competition among users makes it impossible for a user to excessively increase its congestion window size in order to improve its own quality. The increase of the congestion window size \( W_n^k \) may decrease the priority metrics and hence, reduce the resulting distortion reduction in the next time slot. Following Remark 4 in Section IV, the multimedia distortion reduction of user \( V_n \) in the next time slot can be written as

\[
\Delta Q_n^{k+1}(W_n^k) = \sum_{\forall CL_{mn} \in V_n} Q_{mn}N_{mn}I(E[P_{mn}^{k+1}(W_n^k)] > 0).
\]  (13)

where \( Q_{mn} \) represents the distortion impact of class \( CL_{mn} \). Comparing \( \Delta Q_n^{k+1}(W_n^k) \) with the current multimedia distortion reduction \( Q_n^k \), the variation appears only for the classes whose priority metrics change sign. If we denote \( M_n^k \) as the set of classes whose priority metrics follow \( PM_{mn}^k E[P_{mn}^{k+1}] < 0 \), we can rewrite equation (13) as

\[
\Delta Q_n^{k+1} = \begin{cases} Q_n^k, & \text{if } M_n^k = \emptyset \\ Q_n^k + \Delta Q_n^k, & \text{if } PM_{mn}^k < E[P_{mn}^{k+1}], \forall CL_{mn} \in M_n^k \\ Q_n^k - \Delta Q_n^k, & \text{if } PM_{mn}^k > E[P_{mn}^{k+1}], \forall CL_{mn} \in M_n^k \end{cases} 
\]  (14)

where \( \Delta Q_n^k = |Q_n^{k+1} - Q_n^k| = \sum_{\forall CL_{mn} \in M_n^k} Q_{mn}N_{mn} \geq 0 \) represents the difference of the expected distortion reduction of user \( V_n \) in time slot \( k \). Based on equation (14), we next prove the sufficient condition for achieving the discussed quality-based fairness.

**Lemma 2:** The difference of the fairness index is nonnegative, i.e., \( \Delta F^k = F^{k+1} - F^k \geq 0 \), if

\[
\sum_{V_n \in V} \left( \frac{Q_n^k}{F_n^k} \right)^2 \sum_{V_n \in V} \Delta Q_n^k \geq \sum_{V_n \in V} F_n^k \sum_{V_n \in V} F_n^k \Delta Q_n^k. 
\]  (15)

**Proof:** We omit the proof here due to space limitations. A similar proof can be found in [16].

Lemma 2 provides a sufficient condition that ensures a nondecreasing fairness index \( \Delta F^k \). Since the index is bounded by 1, the interaction among users asymptotically drives the fairness index to 1 [16]. Based on Lemma 2, the following Theorem provides the sufficient conditions for multiple myopic media-TCP users to reach the quality-based fairness.

**Theorem 4:** The fairness index of multiple media-TCP users \( V_n \in V \) converges to 1, i.e., \( F^\infty = 1 \), if the following sufficient conditions are satisfied:

1. \( Q_{mn} = Q_{mn}^\prime = Q_m \forall V_n, V_n^\prime \in V \)
2. \( N_{mn} \geq N_{mn}^\prime \) for any \( Q_m \geq Q_m^\prime, \forall V_n, V_n^\prime \in V \).
3. \( E[P_{mn}^{k+1}(W_n^k)], \forall CL_{mn} \in V_n, \forall V_n \in V \) are all nonincreasing functions of \( W_n^k \) using the same \( \lambda \).

**Proof:** See Appendix D.

The first condition in Theorem 4 indicates that the multimedia-TCP users apply the same set of \( [Q_m, m = 1,..., M] \) to classify their \( M \) classes of the multimedia applications. The second condition indicates that for all the users in the network, users always have more packets in a class with higher distortion impact than a class with lower distortion impact. This condition is commonly seen in many video coding techniques. For example, in MPEG video frames, I-frames usually contain much more information bits than P-frames and B-frames. The third condition is discussed in Lemma 1. Based on equation (14), as long as the priority metrics \( E[P_{mn}^{k+1}(W_n^k)] \) are nonincreasing functions of \( W_n^k \) for all the classes, we can show that a user with a larger multimedia quality \( Q_n^k \) will always have a smaller quality change \( \Delta Q_n^k \) when using media-TCP to change their congestion window size. In Appendix D, we prove that this allows the proposed media-TCP to satisfy the sufficient condition in Lemma 2 and hence the quality-based fairness index converges to 1. Finally, the three conditions in Theorem 4 lead to \( F^\infty = 1 \) for media-TCP users \( V_n \in V \).

**VI. SIMULATION RESULTS**

In this section, we simulate the proposed congestion control scheme using different video sequences: “Forman”, “Mobile”, and “Coastguard” (at a frame rate of 30 Hz, CIF format). The sequences are encoded using an embedded scalable video codec [20] at the bitrate of 1500Kbps. We assume that each Group of Picture (GOP) contains 16 frames and each of them can tolerate a playback delay of \{133, 266, 400, 533\} ms. We set the packet length up to 1000 bytes and the video packets are classified into sixteen classes based on their spatial and temporal interdependencies as in [19][20]. Table II provides a summary of the classifications of the sequences. We simulate the video transmission using MATLAB using the simulation settings in Figure 4. There are 20 regular TCP users and the resulting average RTT for the video packets is 133ms.

A. Tradeoff between multimedia performance and TCP friendliness
First, we simulate the case without the media-TCP user 2. We focus on the media-TCP user 1 streaming the “Coastguard” sequence using the proposed media-TCP congestion control with different Lagrangian multipliers and discount factors. Based on the measured packet loss rate, the media-TCP user applies a Markov chain model (similar to the model applied in [9]) on the network states with an expected TCP window size $W_{TCP} = 16$ per RTT, and the horizon $K = 4$ RTT. Figure 5 shows the tradeoff between multimedia quality and the horizon-$K$ TCP-friendliness. Larger $\lambda$ provides better TCP-friendliness, but achieves lower multimedia quality, because the quality gain is weighed less than the cost within the instantaneous utility. The results also show that the foresighted approach with larger $\gamma$ significantly improves the multimedia quality while maintaining moderate TCP-friendliness. However, in this paper, we focus on deriving the optimal solution when the environment (i.e. the state transition probabilities) and the utility are perfectly known. If the transition probabilities are not perfect, a larger $\gamma$ can lead to a worse learning performance. The selection of $\lambda$ and $\gamma$ for media-TCP using online learning for the case when the environment is unknown represents an interesting future research direction.

| Class $CI_{mn}$ | 1 | 2 | 3 | 4 | 5–8 | 9–16 |
|----------------|---|---|---|---|-----|------|
| $Q_m$ (dB/ pkt) range | 0.154 | 0.153 | 0.09 | 0.08 | -0.072 | -0.053 |
| $N_{m}^0$ of “Coastguard” per GOP | 17 | 17 | 12 | 12 | 5 | 4 |
| $N_{m}^0$ of “Forman” per GOP | 34 | 34 | 8 | 8 | 4 | 0 |
| $N_{m}^0$ of “Mobile” per GOP | 30 | 30 | 13 | 13 | 1 | 0 |

Table II. Classification of the Sequences.

Fig. 5 (a) Average Y-PSNR of Coastguard sequence, (b) resulting TCP-friendliness versus different Lagrangian multipliers (playback delay: 266ms, $K = 4$ RTT).

B. Comparisons against alternative congestion control solutions for multimedia applications

We simulate separately the streaming of “Coastguard” sequence as well as “Forman” sequence using three different approaches: 1) our proposed packet-based media-TCP congestion control (MT) that jointly optimizes the transmission scheduling and congestion window size; 2) a flow-based rate-distortion optimization approach (RD) [11] to optimize the transmission scheduling by adapting the sending rate to the available TCP throughput; 3) passive multimedia transmission directly over TCP connections (PA) as in [5].

Figure 6 shows the average video quality of various approaches using different playback delays and clearly demonstrates that the joint transmission scheduling and congestion control optimization is essential for real-time multimedia transmission. Our proposed approach significantly outperforms the others especially when the playback delay is smaller than 400ms (which is common in numerous real-time video streaming and videoconferencing applications), because it is able to jointly optimize the congestion window size as well as the transmission scheduling by considering the distortion impacts, delay deadlines, and interdependencies of the packets.

C. Fairness among multiple media-TCP streams

In this subsection, we validate the quality-based fairness among multiple users using media-TCP. We simulate the case when multimedia user 1 streams “Coastguard” sequence and multimedia user 2 streams “Mobile” sequence simultaneously using the same simulation settings in Figure 4 with 20 regular TCP users. The playback delay is set as 533ms. Figure 7 shows the congestion window size and the video quality over time when the multimedia users apply the passive multimedia transmission approach (PA) directly over TCP connections as in [5]. It is shown that although the congestion window sizes of the multimedia users follow the average TCP window size of the 20 regular TCP users, the video quality gap between the two sequences is always larger than 3 dB.

On the other hand, Figure 8 shows the congestion window size and the video quality over time when the multimedia users apply the proposed media-TCP congestion control (MT) algorithms. We classify the video packets of both sequences to satisfy the first two sufficient conditions in Theorem 4. We also set a small discount factor $\gamma = 0.1$ to ensure that the priority metrics $E[PM_{mn}^{k+1}(W_{mn}^k)]$ are nonincreasing functions
III. It is shown that the proposed media-TCP congestion window sizes and the video qualities are summarized in Table using the same simulation settings. The resulting congestion resources in terms of video quality.

We further increase the number of the regular TCP users and the average congestion window size of the multimedia users using PA approach (playback delay: 533ms).

Fig. 7 (a) Congestion window sizes over time for the two multimedia users and the average congestion window size of the 20 regular TCP users. (b) Video quality over time for the two multimedia users using PA approach (playback delay: 533ms).

Fig. 8 (a) Congestion window sizes over time for the two multimedia users using MT approach (playback delay: 533ms). (b) Video quality over time for the two multimedia users using MT approach (playback delay: 533ms).

sequences as the number of users in the network increases, while having limited impact on the other regular TCP users. This is because the media-TCP is able to better utilize the resource by prioritizing the video classes for transmission, in addition to merely adapting the congestion window size.

VII. CONCLUSIONS

In this paper, we formulate a media-aware congestion control for multimedia transmission using FHMDP that explicitly considers the distortion impacts, delay deadlines and interdependencies of the various multimedia classes. The proposed approach not only adapts the congestion window size given the measured packet loss rate, but also optimally prioritizes the multimedia classes for transmission and hence, further improves the multimedia quality. We show that this complex FHMDP problem can be decomposed into simpler optimal stopping problems, thereby significantly reducing the complexity of solving the problem. The simulation results show that the proposed foresighted media-TCP significantly outperforms the conventional TCP-friendly congestion control schemes in terms of quality, especially for real-time streaming with a small playback delay. Moreover, unlike the conventional congestion control approaches focusing on the throughput-based fairness, our solution maintains the quality-based fairness among the multimedia users, which improves the overall streaming quality by utilizing the available bandwidth resources more efficiently.

Table III. The comparisons of the video qualities using the PA approach and the proposed MT approach when the number of users in the network increases.

| (playback delay: 533ms) | PA approach | MT approach (λ = 10, γ = 0.1, K = 4 RTT) |
|------------------------|-------------|---------------------------------|
| Number of TCP users    |             |                                 |
| N = 20                 | Avg. PSNR  | Avg. PSNR  |
|                        | of user 1 (dB) | of user 2 (dB) |
|                        | Avg. window size of TCP users | Avg. window size of TCP users |
|                        | Avg. PSNR of user 1 (dB) | Avg. PSNR of user 2 (dB) |
|                        | Avg. PSNR of the users (dB) | Avg. PSNR of the users (dB) |
|                        | Avg. window size of TCP users | Avg. window size of TCP users |
| N = 25                 | 32.07       | 27.46 |
|                        | 29.76       | 17.00 |
|                        | 31.58       | 30.43 |
|                        | 31.00       | 16.95 |
| N = 30                 | 29.88       | 23.38 |
|                        | 26.63       | 11.70 |
|                        | 30.29       | 30.22 |
|                        | 11.51       | 11.51 |

Algorithm 1 Media-TCP congestion control with independent packets

For time slot $k = i$, given the current state $s^*, \lambda, \gamma, K$

Set $k = i + K - 1$

While $k \geq i$

For all classes $CL_m$

Compute all $J^{*}_m(W^{TCP}_m, N^{k}_m)$ from equation (17), for $N^{k}_m \in \{0,...,N^{\max}_m\}$, $W^{TCP}_m \in \{0,...,W^{\max}_m\}$

Compute $PM^{k^*}_m(s^k)$ and $\pi^k_m(s^k)$ in equation (10);

End for

Set $k = k - 1$

End while

Set $PM^{*}_m = PM^{k^*}_m(s^k)$ for $m = 1,...,M$

Set $W^{*} = \sum_{\forall m:PM^{*}_m \geq 0} N^{*}_m$;
Algorithm 2: Media-TCP congestion control with interdependent packets

For time slot $k = i$, given the current state $s^i, \lambda, \gamma, K$

Set $k = i + K - 1$;
While $k \geq i$
  Set $W^k = 0$ and $\text{depth} = 0$;
  For all classes $CL_m$ with $\text{depth}_m = \text{depth}$
    Compute all $J_m^k(W^k_{TCP}, N_m^k)$ from equation (17),
    for $N_m^k \in \{0, ..., N_m^{\text{max}}\}, W^k_{TCP} \in \{0, ..., W^k_{\text{TCP}}\}$;
    Compute $PM^k_{m}(s^k)$ and $\pi^*_m(s^k)$ in equation (11);
    If $PM^k_{m}(s^k) < 0$, set $Q_m = 0$ for $\forall CL_m \in \text{Des}_m$;
    Set $depth = depth + 1$;
  End for
Set $k = k - 1$;
End while
Set $PM^*_m = PM^k_{m}(s^i)$ for $m = 1, ..., M$;
Set $W^* = \sum_{\forall m, PM^*_m \geq 0} N_m^k$;

APPENDIX B

Proof of Theorem 1: Without losing generality, we assume $i = 1$. Since packets are independent, the distortion reduction $Q^k(a^i, s^i)$ in equation (6) can be computed by

$$Q^k(a^i, s^i) = \sum_{m=1}^{M} Q_m N_m^k \pi_m^k, \quad \text{First, we see that when } k = K,$$

$$J^k_{\mu}(s^k) \text{ can be rewritten as:}$$

$$J^k_{\mu}(s^k) = \sum_{m=1}^{M} \left( Q_m - \frac{\lambda}{W^k_{\text{TCP}}} \right) N_m^k \pi_m^k + \frac{\lambda M}{W^k_{\text{TCP}}}$$

where $\pi^*_m, m = 1, ..., M$ is $\mu(s^k)$ denotes the vector of transmission permissions given the policy $\mu(s^k)$. Based on this, the expected utility-to-go $J^k_{\mu}(s^k) = \sum_{m=1}^{M} \left( Q_m - \frac{\lambda}{W^k_{\text{TCP}}} \right) N_m^k \pi_m^k + \frac{\lambda M}{W^k_{\text{TCP}}}$ is separable and also a nondecreasing function of $N_m^k$. Then, by assuming $J_{\mu}^{k+1}(s^{k+1}) = \sum_{m=1}^{M} J_{\mu,m}^k(W^k_{TCP}, N_m^k) + C^{k+1}(W^k_{TCP})$ and assume $J_{\mu}^{k+1}(s^{k+1})$ are nondecreasing functions of $N_m^{k+1}$ for all classes, we have

$$J^k_{\mu}(s^k) = u(s^i, \mu(s^i)) + \sum_{s^{k+1} \in S} P(s^{k+1} | s^k) J^k_{\mu}(s^{k+1})$$

$$= \sum_{m=1}^{M} \left( Q_m - \frac{\lambda}{W^k_{\text{TCP}}} \right) N_m^k \pi_m^k + \frac{\lambda M}{W^k_{\text{TCP}}} + \gamma \sum_{s^{k+1} \in S} P(W^k_{TCP} | W^k_{TCP}) + C^{k+1}(W^k_{TCP})$$

APPENDIX C

Proof of Theorem 2: From equation (9), the optimal policy in time slot $i$ with state $s^i$ is the action that maximize the expected utility. Based on Theorem 1, we can see that when the packets in the buffer are independent, the utility-to-go is separable. From equation (17), the optimization problem becomes

$$[\pi^*_1(s^i), ..., \pi^*_M(s^i)] = \text{arg max}_{\pi_m^* \in \pi^*_m} \left\{ \sum_{m=1}^{M} \left( Q_m - \frac{\lambda}{W^k_{\text{TCP}}} \right) N_m^k \pi_m^k + \frac{\lambda M}{W^k_{\text{TCP}}} \right\}.$$

In other words, we have

$$J^i_{\mu,m}(W^k_{TCP}, N_m^i) =$$

$$J_1 = \left\{ Q_m - \frac{\lambda}{W^k_{\text{TCP}}} \right\} N_m^i + \gamma \sum_{s^{k+1} \in S} P(W^k_{TCP} | W^k_{TCP}) J^i_{\mu,m}(W_{TCP}, N_m^i) \text{ if } \pi_m^i = 1$$

$$J_0 = \gamma \sum_{s^{k+1} \in S} P(W^k_{TCP} | W^k_{TCP}) J^i_{\mu,m}(W_{TCP}, N_m^i - N_{m}^{D_i} + N_{m}^{A_i}) \text{ if } \pi_m^i = 0$$

Hence, by defining $PM^*_m(s^i) = J_1 - J_0$ in equation (18), the problem becomes an optimal stopping problem [23] that leads to the utility-to-go in equation (9).

APPENDIX D

Proof of Theorem 4: Let $|V| = 1$, the sufficient condition in equation (15) is satisfied. Without loss of generality, let $Q_1 \geq Q_2 \geq ... \geq Q_h \geq Q_{h+1} = Q$. Suppose that the condition is satisfied when $|V| \leq h$, i.e.
\[
\sum_{n=1}^{h} (\bar{Q}_n)^2 \sum_{n=1}^{h} \Delta \bar{Q}_n \geq \sum_{n=1}^{h} \bar{Q}_n \sum_{n=1}^{h} \bar{Q}_n \Delta \bar{Q}_n .
\] (19)

For \(|V| = h + 1\), since \(Q \leq \bar{Q}_n, \forall V_n \in V\), if the sufficient conditions are satisfied, we have priority metrics
\[
\Delta Q = \sum_{\forall n \in M_k} Q_n M_{n,h+1} \geq \sum_{\forall n \in M_k} Q_n M_{n,m} = \bar{Q}_n, \forall V_n \in V
\]

Hence, we have
\[
\sum_{n=1}^{h} \Delta Q \bar{Q}_n (\bar{Q}_n - Q) \geq \sum_{n=1}^{h} \Delta Q \bar{Q}_n (\bar{Q}_n - Q)
\]
\[
\Rightarrow Q^2 \sum_{n=1}^{h} \Delta Q \bar{Q}_n + \Delta Q \sum_{n=1}^{h} \bar{Q}_n \geq \sum_{n=1}^{h} \bar{Q}_n \Delta \bar{Q}_n + Q \Delta Q \sum_{n=1}^{h} \bar{Q}_n
\]
\[
\Rightarrow h \sum_{n=1}^{h} \bar{Q}_n + Q \left( \sum_{n=1}^{h} \bar{Q}_n \Delta \bar{Q}_n + Q \Delta Q \right) \geq \sum_{n=1}^{h} \bar{Q}_n \Delta \bar{Q}_n + Q \Delta Q \sum_{n=1}^{h} \bar{Q}_n
\]
Adding equation (19) and (20), we have,
\[
\left( \sum_{n=1}^{h} \bar{Q}_n \right)^2 \geq \sum_{n=1}^{h} \Delta \bar{Q}_n + Q \Delta Q \sum_{n=1}^{h} \bar{Q}_n
\]
which satisfy the condition in equation (15). By induction, we prove that the sufficient condition of the lemma is fulfilled for all \(|V| \geq 1\).

REFERENCES

[1] J. Mo and J. Walrand, “Fair End-to-End Window-Based Congestion Control,” IEEE/ACM Transactions on Networking, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[2] J. Padhye, V. Firoiu, D. F. Towsley, J. F. Kurose, “Modeling TCP Reno Performance: a simple model and its empirical validation,” IEEE/ACM Trans. on Networking, vol. 8, no. 2, pp. 133-145, Apr. 2000.

[3] J. Padhye, V. Firoiu, D. F. Towsley, J. F. Kurose, “A Model Based TCP-Friendly Rate Control Protocol,” In Proc. of NOSSDAV’99, 1999.

[4] S. Floyd, M. Handley, J. Padhye, J. Widmer, “Equation-Based Congestion Control for Unicast Applications,” ACM SIGCOMM Computer Communication Review, vol. 30, no. 4, pp. 43-56, Oct. 2000.

[5] B. Wang, J. Kurose, P. Shenoy, D. Towsley, “Multimedia Streaming via TCP: An analytic Performance Study,” ACM Trans. on Multimedia Computing, Communications, and Applications, vol. 4, no. 2, May 2008.

[6] S. Bohacek, “A Stochastic Model of TCP and Fair Video Transmission,” IEEE INFOCOM 2003.

[7] I. V. Bajic, O. Tickoo, A. Balan, S. Kalyanaraman, and J. W. Woods, “Integrated end-to-end buffer management and congestion control for scalable video communications,” Proc. IEEE ICIP 2003, Barcelona, Spain, vol. 3, pp. 257–260, Sept. 2003.

[8] W. Tan and A. Zakhor, “Real-time Internet Video Using Error Resilient Scalable Compression and TCP-Friendly Transport Protocol,” IEEE Trans. on Multimedia, vol. 1, no. 2, June 1999.

[9] T. Nguyen, A. Zakhor, “Distributed video streaming with forward error correction,” In Proc. of Int. Packet Video Workshop, 2002.

[10] H. Seferoglu, U. C. Koçat, M. R. Civanlar, and J. Kempf, “Congestion State-Based Dynamic FEC Algorithm for Media Friendly Transport Layer,” In Proc. of Int. Packet Video Workshop, 2009.

[11] Q. Zhang, W. Zhu, Y. Zhang, “Resource Allocation for Multimedia Streaming over the Internet,” IEEE Trans. on Multimedia, vol. 3, no. 3, Sep. 2001.

[12] P. Zhu, W. Zeng, C. Li, “Joint Design of Source Rate Control and QoS-Aware Congestion Control for Video Streaming over the Internet,” IEEE Trans. on Multimedia, vol. 9, no. 2, pp. 366-376, Feb. 2007.

[13] R. Rejaie, M. Handley, and D. Estrin, “RAP: An End-to-End Rate-based Congestion Control Mechanism for Real-time Streams in the Internet,” IEEE INFOCOM 1999.

[14] L. Cai, X. Shen, J. Pan, and J. W. Mark, “Performance Analysis of TCP-Friendly AIMD Algorithms for Multimedia Applications,” IEEE Trans. on Multimedia, vol. 7, no. 2, pp. 339-355, Apr. 2005.

[15] G. Wu, E. K. P. Chong, R. Givan, “Burst-Level Congestion Control Using Hindsight Optimization,” IEEE Trans. on Automatic Control, vol. 47, no. 6, pp. 979-991, June 2002.

[16] N. R. Sastry and S. S. Lam, “CYRF: A Theory of Window-Based Unicast Congestion Control,” IEEE/ACM Trans. Networking, vol. 13, no. 2, pp. 330-342, Apr. 2005.

[17] P. Chou, and Z. Miao, “Rate-distortion optimized streaming of packetized media,” IEEE Trans. Multimedia, vol. 8, no. 2, pp. 390-404, 2005.

[18] F. Fu, M. van der Schaar, “Structural solutions of cross-layer optimization of wireless multimedia transmission,” Technical Report, 2009. http://medianetlab.ee.ucla.edu/papers/UCLAtechReport.CLO.pdf

[19] H. Shiang and M. van der Schaar, "Multi-user video streaming over multi-hop wireless networks: A distributed, cross-layer approach based on priority queuing," IEEE J. Sel. Areas Commun., vol. 25, no. 4, pp. 770-785, May 2007.

[20] J.-R. Ohm, M. van der Schaar, and J. W. Woods, "Interframe wavelet coding x motion picture representation for universal scalability," EURASIP Signal Processing: Image Communication, Special issue on Digital Camera, vol. 19, no. 9, pp. 877-908, Oct. 2004.

[21] T. Wiegand, G. J. Sullivan, G. Bjontegaard, A. Luthra, “Overview of the H.264/AVC video coding standard,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, July, 2003.

[22] M. Dai, D. Loguinov, and H. Radha, “Rate distortion modeling for scalable video coders,” In proc. of ICIP, 2004.

[23] D. F. Bertsekas, Dynamic programming and Optimal Control, 2 volume set, Athena Scientific, 1995.

[24] D. S. Turaga and T. Chen, “Hierarchical Modeling of Variable Bit Rate Video Sources,” Packet Video, 2001.

[25] Y. Li, Z. Li, M. Chiang, and A. R. Calderbank, “Content-aware distortion fair video streaming in networks,” IEEE Trans. Multimedia, to appear.

[26] R. Jain, D.-M. Chiu, and W. Hawe, “A quantitative measure of fairness and discrimination for resource allocation in shared computer systems,” Digital Equipment Corporation, DEC Research Report, Tech. Rep. TR-301, Sep. 1984.

[27] P. A. Chou, “Streaming media on demand and live broadcast,” Multimedia over IP and Wireless Networks: Compression, Networking, and Systems, Academic Press, 2007.

[28] E. Kohler, M. Handley, and S. Floyd. Datagram Congestion Control Protocol (DCCP). Proposed standard RFC 4340, IETF, http://www.ietf.org/rfc/rfc4340, March 2006.

[29] Y. Li, M. Chiang, and A. R. Calderbank, “Congestion control in networks with delay sensitive traffic,” Proc. IEEE GLOBECOM, Washington D.C., Nov. 2007.
**Table IV. Nomenclature**

| $Q^k_m$ | Distortion impact of the class $CL^k_m$. | $N^k$ | $N^k = [N^k_m, m = 1, \ldots, M]$ |
|---------|------------------------------------------|-------|-----------------------------------|
| $N^k_m$ | Number of packets in the transmission buffer of class $CL^k_m$ in time slot $k$ | $a^k$ | Action at time slot $k$ of the FHMDP problem: $a^k = [a^k_m, m = 1, \ldots, M] \in \{0, 1\}^M$ |
| $N^k,A_m$ | Number of packets in class $CL^k_m$ that arrive in time slot $k$ | $s^k$ | State at time slot $k$ of the FHMDP problem: $s^k = \{W^k_{TCP}, N^k\} \in S$ |
| $N^k,D_m$ | Number of packets in class $CL^k_m$ whose delay deadline is expired in time slot $k$ | $\lambda$ | Positive Lagrangian multiplier of the FHMDP problem |
| $depth^k_m$ | Maximum distance from class $CL^k_m$ to the root in the DAG in time slot $k$ | $\gamma$ | Discount factor of the FHMDP problem |
| $\bar{Q}^k$ | Expected multimedia distortion reduction in time slot $k$ | $K$ | Finite horizon of the FHMDP problem |
| $PM^k_m$ | Priority metric of class $CL^k_m$ in time slot $k$ | $J^k(s^k)$ | Expected utility-to-go in time slot $k$ for the FHMDP problem |
| $x^k$ | $x^k = [PM^k_1, \ldots, PM^k_M] \in \mathbb{R}^M$ | $J^k_m(W^k_{TCP}, N^k_m)$ | Expected utility-to-go component specific to the class $CL^k_m$ |
| $\pi^k_m$ | Transmission permission for transmitting the packets in class $CL^k_m$ in time slot $k$ | $\mathcal{V}$ | A set of media-TCP users in the network, $\mathcal{V} = \{V_n, n = 1, \ldots, V\}$ |
| $W^k$ | Congestion window size in time slot $k$ | $\mathcal{F}^k$ | Jain’s fairness index of the media-TCP users in the network |
| $p^k$ | Estimated packet loss rate in time slot $k$ | $\bar{Q}^k_n$ | User $V_n$’s expected multimedia distortion reduction in time slot $k$ |
| $W^k_{TCP}(p^k)$ | Expected TCP window size (network congestion metric) | $\Delta \bar{Q}^k_n$ | Expected distortion reduction variation of user $V_n$ in time slot $k$, $\Delta \bar{Q}^k_n = |\bar{Q}^k_{n+1} - \bar{Q}^k_n|$ |