Structural Solutions for Dynamic Scheduling in Wireless Multimedia Transmission

Fangwen Fu, *Member, IEEE,* and Mihaela van der Schaar, *Fellow, IEEE*

Abstract—In this paper, we propose a systematic solution to the problem of scheduling delay-sensitive media data for transmission over time-varying wireless channels. We first formulate the dynamic scheduling problem as a Markov decision process that explicitly considers the users' heterogeneous multimedia data characteristics (e.g., delay deadlines, distortion impacts and dependences, and so on) and time-varying channel conditions, which are not simultaneously considered in state-of-the-art packet scheduling algorithms. This formulation allows us to perform foresighted decisions to schedule multiple data units for transmission at each time in order to optimize the long-term utilities of the multimedia applications. The heterogeneity of the media data enables us to express the transmission priorities between the different data units as a priority graph, which is a directed acyclic graph. This priority graph provides us with an elegant structure to decompose the multidata unit foresighted decision at each time into multiple single-data unit foresighted decisions which can be performed sequentially, from the high priority data units to the low priority data units, thereby significantly reducing the computation complexity. When the statistical knowledge of the multimedia data characteristics and channel conditions is unknown *a priori*, we develop a low-complexity online learning algorithm to update the value functions, which capture the impact of the current decision on the future utility. The simulation results show that the proposed solution significantly outperforms existing state-of-the-art scheduling solutions.

Index Terms—Delay sensitive scheduling, directed acyclic graph (DAG), energy-efficient multimedia transmission systems, multimedia streaming.

I. Introduction

EXISTING wireless networks provide dynamically varying resources with only limited support for the quality of service required by delay-sensitive, bandwidth-intense and loss-tolerant multimedia applications. One of the key challenges for delivering the multimedia data over wireless networks is the *dynamic* characteristics of both the wireless channels and the multimedia source data [1]. To overcome this challenge, packet scheduling optimization has been extensively investigated in recent years in order to maximize the quality of the multimedia applications given the underlying resource constraints.

The multimedia data is often encoded and encapsulated into multiple data units (DUs), which can be video frames, slices, packets, and so on. Different DUs often have different distortion impacts, delay deadlines, and dependences. Existing packet scheduling solutions often ignore these heterogeneous characteristics of multimedia applications. For example, [2]–[5] only assumed the upper layer packets are homogeneous and share the same delay deadline. References [8], [9], and [25] did not take into account the complicated dependences among media packets. In [26], a channel, deadline, and distortion-aware packet scheduling algorithm is developed, where only one frame is considered for transmission at each time and an i.i.d. channel is assumed. In this paper, the dependences among video frames are not considered. In [13], the packet scheduling is optimized based on packets' hard delay deadlines, distortion impacts, and the underlying time-varying wireless channel. However, [13] did not explicitly take into account the dependences among packets. To take into account the dependences among packets, in [7], [10], and [12], the packet scheduling is optimized using a rate-distortion framework (named RaDiO), which expresses the dependences among packets as a directed acyclic graph (DAG). However, RaDiO disregards the time-varying characteristics of the considered transmission network, thereby leading to a suboptimal performance over wireless networks. In [6] and [11], the scheduling of video packets over a time-varying wireless channel is formulated as a cross-layer optimization problem. However, these cross-layer optimization solutions only maximize the quality of the currently transmitted video packets based on the observed channel conditions, without considering future transmission opportunities and the impact of current decisions on the long-term video quality. This type of optimization will be referred to in our paper as myopic optimization.

In summary, a systematic packet scheduling optimization framework for media communication that explicitly considers both the heterogeneous characteristics of the multimedia traffic and the time-varying wireless conditions is still missing. To overcome this challenge, in this paper we develop a systematic energy-aware packet scheduling framework for single-user multimedia transmission over a time-varying wireless channel.

To capture the heterogeneous characteristics of DUs, we first introduce the concept of a "context" at each time slot to represent the heterogeneity of the DUs available for

F. Fu was with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA. He is now with Intel, Folsom, CA 95630 USA (e-mail: fangwen.fu@intel.com).

M. van der Schaar is with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA (e-mail: mihaela@ee.ucla.edu).

TABLE I

SUMMARY OF MAIN NOTATIONS

Notation	Description	Notation	Description
f_j^g	jth DU in GOP g	$l_{f_j^g}$	Size of DU f_j^g
$q_{f_j^g}$	Distortion impact of DU f_j^g	$d_{f_j^g}$	Delay deadline of DU f_j^g
$f_{j'}^g \prec f_j^g$	DU f_j^g depends on DU $f_{j'}^g$	W	Scheduling time window
$y_{f,t}$; $\mathbf{y}_t$	Transmitted data of DU f at time t; transmitted data of all DUs in context C_t	$x_{f,t}$; $\mathbf{x}_t$	Remaining packets of DU f at time t; remaining packets of all DUs in context C_t
C_t	Context at time t	T_t	Traffic state
h_t	Channel state at time t	s_t	State including both traffic state and channel state
$u\left(s_t, \mathbf{y}_t\right)$	Utility at state s_t when $\mathbf{y}_t$ amount of data is transmitted	$\rho(h, y)$	Energy function
α	Discounted factor	$V(C_t, \mathbf{x}_t, h_t)$	State value function
$U(C_t, \mathbf{z}_t, h_t)$	Postdecision state value function	$f \lhd f'$	DU f has higher transmission priority than DU f'
PG_t	Priority graph	$U_f\left(C_t, x_{f,t}, h_t\right)$	Postdecision state value function for DU f
$p_f(z)$	Dependence factor	λ	Tradeoff parameter

transmission at each time slot. Through the context concept, we are able to capture the dynamic features of the multimedia packets across time. We then formulate the dynamic packet scheduling optimization as a Markov decision process (MDP) problem [14] by further considering the underlying channel dynamics. Within the MDP formulation, the packet scheduling is performed in a foresighted fashion in order to maximize the long-term reconstructed multimedia quality.

In the conventional MDP formulation, the foresighted decision for the packet scheduling is often coupled with the expectation over the experienced dynamics, which makes the packet scheduling problem hard to solve in unknown environments (i.e., where the statistical knowledge of the multimedia data arrivals and channel state transitions is unknown). To resolve this obstacle, we introduce a postdecision state (which is a "middle" state, in which the transmitter finds itself after packet transmission but before the new packet arrivals and new channel realization) and a corresponding postdecision state-value function which represents the optimal long-term utility starting from the postdecision state. Through the postdecision state value function, we can *separate* the foresighted decision on the packet scheduling from the expectation over the underlying dynamics. In other words, the foresighted decision can be computed without knowing the experienced dynamics, given the postdecision state value function. The postdecision state value functions can then be updated online accordingly. Hence, the postdecision state concept allows us to separate the packet scheduling at each time slot into two phases (i.e., two-phase packet scheduling): one is the foresighted decision on the optimal scheduling given the postdecision state value function, and the other one is the online update on the postdecision state value function.

In order to reduce the complexity involved in computing the packet scheduling policy, we define the transmission priorities of the DUs in each context based on the distortion impacts, delay deadlines and dependences, and express them as a DAG, which we refer to simply as the priority graph. Different from

the DAG expression on the source coding dependences in [7], the proposed DAG construction represents the transmission priorities which include the packet dependences. Through the priority graph, we are able to *separate* the multi-DU foresighted decision at each time slot into multiple single-DU foresighted decisions and the two-phase packet scheduling is applied to each individual DU, which significantly reduce the complexity in computing the optimal foresighted decisions.

In the unknown environment, we further develop an online learning algorithm to estimate the postdecision state value functions. Based on the separation developed for multi-DU foresighted optimization, we are able to estimate the postdecision state value functions for each DU using a low-complexity online learning method.

This paper is organized as follows. Section II characterizes the attributes of the multimedia traffic. Section III formulates the packet scheduling problem for multiple independently decodable DUs as an MDP and develops structural solutions to determine the optimal packet scheduling policies. Section IV further extends the structural results to the packet scheduling for interdependent DUs. Section V presents the simulation results, followed by the conclusions in Section VI. A summary of main notations is listed in Table I.

II. MULTIMEDIA TRAFFIC CHARACTERISTICS

In this section, we discuss how the heterogeneous attributes of multimedia traffic[1] can be represented. In past work, multimedia traffic (e.g., video traffic) is often modeled as a leaky bucket with constraints (e.g., peak rate constraint, average delay constraint, and so on) [19]. However, this model only characterizes the rate change in multimedia traffic and does not explicitly consider the heterogeneous characteristics of multimedia data. In this section, we aim to develop a general model to represent the encoded multimedia data with heterogeneous characteristics (e.g., various delay deadlines, distortion

---

[1]Multimedia traffic can be generated in real time or be pre-encoded.

impacts, dependences, and so on). Using this multimedia traffic model, we will be able to dynamically optimize packet scheduling for multimedia transmission over time-varying wireless networks, which is presented in Sections III and IV.

A. Attributes of Data Units

In this section, we discuss how the heterogeneous attributes of the multimedia data can be modeled. The multimedia data is often encoded periodically using a group of pictures (GOP) structure, which lasts a period of T time slots. The multimedia data within one GOP are encoded interdependently using, e.g., motion estimation, while the data belonging to different GOPs are encoded independently. Note that the prediction-based coding schemes often lead to sophisticated dependences. After being encoded, each GOP contains N data units (DUs), each representing one type of DU (e.g., I, P, B frames in encoded video bitstream) and being indexed by $j \in \{1, \ldots, N\}$. The set of DUs within a GOP $g \in \mathbb{N}$ is denoted by $\left\{ f_1^g, \ldots, f_N^g \right\}$. The attributes of DU f_j^g are listed below.

Size: The size of DU f_j^g is denoted by $l_{f_j^g}$ (measured in packets,[2]) where $l_{f_j^g} \in \left[1, l_j^{\max}\right]$, and $l_j^{\max}$ is the maximum allowable size for the jth DU at each GOP. The size of DU f_j^g is determined when DU f_j^g is encoded. To simplify the exposition, $l_{f_j^g}$ is generated from an i.i.d. random variable[3] with the probability mass function $PMF_{f_j^g}(l)$. Note that $PMF_{f_j^g}(l)$ is the same for the jth DU across different GOPs, but it differs for different types of DUs.

Distortion impact: Each DU f_j^g has a distortion impact $q_{f_j^g}$ per packet, which is assumed to be the same for all the GOPs, i.e., $q_{f_j^g} = q_{f_j^{g'}}, \forall g, g'$. The distortion impact $q_{f_j^g}$ represents the amount by which the multimedia distortion is reduced if one packet from DU f_j^g is received at the decoder side. The distortion impact computation is similar to [6]. Note that the developed framework is also applicable in the case when the packets within one DU have different distortion impact.

Delay deadline: The delay deadline of DU f_j^g represents the time by which the DU should be decoded in order to be displayed. We denote by $d_{f_j^g}$ the delay deadline of DU f_j^g. Since the GOP structure is fixed, the difference between the delay deadlines of the two DUs within one GOP is constant, i.e., $d_{f_j^g} - d_{f_{j'}^g} = \Delta d_{jj'} > 0$ where $j > j'$, and the delay deadlines of the same types of DUs from different GOPs satisfy $d_{f_j^g} - d_{f_j^{g-1}} = T$. In other words, the jth DU periodically appears at each GOP with the period of T time slots, which is the length of one GOP.

Dependence: When one DU f_j^g is encoded based on the prediction from the other DU $f_{j'}^g$, we say that DU f_j^g depends on DU $f_{j'}^g$. Note that the dependences among DUs only occur within one GOP and DUs from different GOPs can be decoded independently (i.e., no dependence between them). The dependences among the DUs within one GOP are expressed as a DAG [7]. The DAG remains the same for a fixed GOP

structure. In this paper, we assume that if DU f_j^g depends on DU $f_{j'}^g$ (i.e., there exists a path directed from DU $f_{j'}^g$ to DU f_j^g in the DAG and denoted by $f_{j'}^g \prec f_j^g$), then $d_{f_j^g} \geq d_{f_{j'}^g}$ and $q_{f_j^g} \leq q_{f_{j'}^g}$. In other words, DU $f_{j'}^g$ should be decoded prior to DU f_j^g and DU $f_{j'}^g$ has higher distortion impact.

B. Traffic State Representation in Each Time Slot

We consider a time-slotted system in which the nth time slot is defined as the time interval $[n\Delta t, (n + 1)\Delta t)$, where Δt is the length of one time slot. In this subsection, we discuss how to represent the multimedia traffic which is ready for transmission at each time slot. At time slot t, as in [7], we assume that the wireless user will only consider for transmission the DUs with delay deadlines in the range of $[t, t + W)$, where W is referred to as the scheduling time window (STW) and assumed to be determined *a priori*.[4] In this paper, we further assume that the STW is chosen to satisfy the following condition: if DU f_j^g directly depends on DU $f_{j'}^g$ (i.e., there is a direct arc from f_j^g to $f_{j'}^g$ in the DAG), then $d_{f_j^g} - d_{f_{j'}^g} < W$. This assumption ensures that DU f_j^g and $f_{j'}^g$ can be considered for transmission at the same time slot.

At time slot t, we introduce a context to represent the set of DUs that are considered for transmission, i.e., whose delay deadlines are within the range of $[t, t + W)$.[5] We denote the context by $C_t = \left\{ f_j^g | d_{f_j^g} \in [t, t + W) \right\}$. Since the GOP structure is fixed, it is easy to show that C_t is periodic with the period of T, which means that for any $f_j^g \in C_t$, there exists $f_j^{g+1} \in C_{t+T}$ and vice versa. Hence, C_t and C_{t+T} have the same types of DUs and the same DAG between these DUs. For example, as shown in Fig. 1, $C_t = \left\{ f_1^g, f_2^g, f_3^g \right\}$ and $C_{t+3} = \left\{ f_1^{g+1}, f_2^{g+1}, f_3^{g+1} \right\}$, where $T = 3$. Since the context represents the set of DUs to be transmitted, it implicitly represents the dependences among the DUs. Due to the periodicity, there are only T different contexts. The transition from context C_t to C_{t+1} is deterministic. It is worthy to know that the context indicates the distortion impacts of the DUs and the dependences among DUs and the context transition indicate the delay deadlines of the DUs.

Given the current context C_t, we let $x_{f,t}$ denote the number of packets in the buffer associated with DU $f \in C_t$. Note that $x_{f,t} \leq l_f$, where l_f represents the amount of the originally available packets for DU f. We denote the buffer state of the DUs in C_t by $x_t = \left\{ x_{f,t} | f \in C_t \right\}$. The traffic state T_t at time slot t is then defined as $T_t = (C_t, x_t)$, where the context represents the types of DUs, the dependences among them, and the buffer state x_t represents the amount of packets remaining for transmission. Hence, the traffic state T_t is able to capture heterogeneous multimedia traffic and is a super-set of existing well-known single-buffer models (which

---

[2]For simplicity, we assume in this paper that each packet has the same length, but this does not affect our proposed solution. It just simplifies our exposition given the space limitations.

[3]The DU size can also be modeled as a random variable depending on previous DUs.

[4]The STW can be determined based on the channel conditions experienced by the user in each time slot. For example, the STW can be set small when the channel conditions are poor (i.e., in the low SNR regime), and large whenever the channel conditions are good (i.e., in the large SNR regime).

[5]We assume that $d_{f_j^g} - t_{f_j^g} \geq W$ which means that the DUs that are considered for transmission at the time slot t must arrive no later than time slot t.
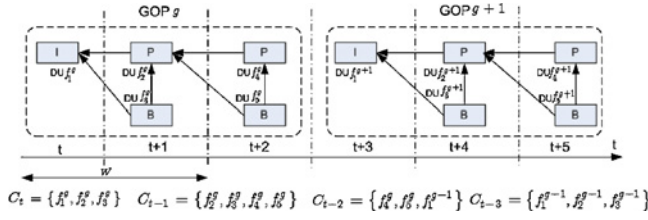
Fig. 1. DAG-based dependences and traffic states at each time slot using IBPBP GOP structure.

ignore packet dependences and delay deadlines) or multibuffer models (which ignore packet dependences *or* delay deadlines).

III. Packet Scheduling for Independently Decodable DUs

We first consider how the packet scheduling optimization should be performed for the DUs that can be independently decoded (e.g., motion JPEG), and consider the interdependent DUs in Section IV.

At each time slot t, the wireless user experiences a channel condition $h_t \in \mathcal{H}$, where $\mathcal{H}$ is the set of finite possible channel conditions and h_t is referred to as the channel state. In this paper, we assume that the wireless channel is slow-fading and the channel condition h_t can be modeled as a finite-state Markov chain [18] with transition probability $p_h\left(h'|h\right) \in [0, 1]$. We further define the state which the wireless user experiences at each time slot t as $s_t = (C_t, x_t, h_t)$, which includes the current context, buffer state and channel state. At time slot t, the wireless user decides how many packets should be transmitted from each DU in the current context. The decision is represented by $y_t\left(C_t, x_t, h_t\right) = \left\{y_{f,t}|f \in C_t\right\}$, where $y_{f,t}$ represents the amount of packets transmitted from DU f and $0 \le y_{f,t} \le x_{f,t}$. We know that the decision in time slot t is a function of the current state s_t. In this paper, we consider the following utility[6] at each time slot t:

$$u\left(s_t, y_t\right) = \sum_{f \in C_t} q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f \in C_t} y_{f,t}\right). \quad (1)$$

In this utility function, the first term represents the distortion reduction obtained by transmitting the amount of data $y_t = \left[y_{f,t}\right]_{f \in C_t}$ from the DUs in the current context. The second term represents the negative value of the consumed energy by transmitting the amount $\sum_{f \in C_t} y_{f,t}$ of packets, where $\lambda > 0$ is the parameter trading-off the distortion reduction and the consumed energy. The energy consumption function $\rho(h, y)$ is assumed to be a convex function of y given the channel condition h. One example is $\rho(h, y) = \sigma^2 \left(e^{2yL} - 1\right)/h$, where is L the length of one packet, which is derived from the information-theoretic rate-power function [20]. In this paper, we only consider the distortion reduction and energy consumption tradeoff instead of the transmission rate constraint because the transmission rate in each time slot is determined by the energy allocation given the channel state.

[6]In this paper, we consider that the multimedia quality is defined as the total distortion reduction of the received media packets.

Then, the wireless user aims to maximize the following long-term expected discounted utility:

$$\max_{y_t(s_t), \forall t} \mathbf{E} \left\{ \sum_{t=0}^{\infty} \alpha^t u\left(s_t, y_t\right) \right\} \quad (2)$$

where $\alpha \in [0, 1)$ is the discount factor. Note that when $\alpha \to 1$, the optimal solution to the optimization in (2) is equivalent to the optimal solution to the problem maximizing the average utility. For independently decodable DUs, given the decision $y_t(s_t)$ in time slot t, the buffer state transition is

$$x_{f,t+1} = \begin{cases} x_{f,t} - y_{f,t}, & \text{if } f \in C_t \bigcap C_{t+1} \\ l_f, & \text{if } f \in C_{t+1} \backslash C_t \end{cases} \quad (3)$$

where the notation "$C_t \bigcap C_{t+1}$" represents the set of DUs persist from time slot t to time slot $t+1$ (i.e., do not expire at the end of time slot t) and $C_{t+1} \backslash C_t$ represents the set of DUs that arrive in time slot $t+1$.

From the above discussion, we know that the channel state transition and buffer state transition are Markovian. We further notice that the buffer state transition also depends on y_t, which is the decision made by the wireless user. Hence, the transition of the state $s_t = (C_t, x_t, h_t)$ is Markovian and the problem in (2) can be formulated as an MDP [14]. In the subsequent sections, we will discuss how the packet scheduling problem can be solved using an MDP formulation.

A. MDP Formulation and Postdecision State-Based Dynamic Programming

In the problem in (2), the decision in each time slot t is to determine the amount of data, $y_{f,t}$ to be transmitted for each DU $f \in C_t$. From [14], we know that the optimal decision can be found by solving the following Bellman's equations:

$$V\left(C_t, x_t, h_t\right) =$$
$$\max_{0 \le y_t \le x_t} \left\{ \begin{array}{l} \sum_{f \in C_t} q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f \in C_t} y_{f,t}\right) + \\ \alpha \mathbf{E}_{h_{t+1}|h_t, l_{t+1}} V\left(C_{t+1}, \left(x_t - y_t\right) \oplus l_{t+1}, h_{t+1}\right) \end{array} \right\}$$
$$(4)$$

where $V\left(C_t, x_t, h_t\right)$ is the state-value function representing the optimal long-term utility starting from the state (C_t, x_t, h_t) and $l_{t+1} = \left\{l_{f'}\right\}_{f' \in C_{t+1} \backslash C_t}$. The operator $z_t \oplus l_{t+1}$ denotes the union operation between $\left\{z_{f,t}\right\}_{f \in C_t \cap C_{t+1}}$ and $\left\{l_{f'}\right\}_{f' \in C_{t+1} \backslash C_t}$, where $\left\{z_{f,t}\right\}_{f \in C_t \cap C_{t+1}}$ represents the remaining data (i.e., data which was not transmitted at time slot t) in DU $f \in C_t \cap C_{t+1}$ after the data transmission at time slot t and $\left\{l_{f'}\right\}_{f' \in C_{t+1} \backslash C_t}$ represents the newly arriving data in DU $f' \in C_{t+1} \backslash C_t$ at time slot $t+1$. It is easy to see that the buffer state fulfils the following condition: $x_{t+1} = \left(x_t - y_t\right) \oplus l_{t+1}$. The expectation in (4) is taken over all the possible new channel states h_{t+1} with the probability of $p\left(h_{t+1}|h_t\right)$ and over the possible data arrival for the DUs in the set $C_{t+1} \backslash C_t$ with the probability of $\prod_{f' \in C_{t+1} \backslash C_t} PMF_{f'}\left(l_{f'}\right)$.

From (4), it is worthy to note that the expectation (over the data arrival and channel state transition) is embedded into the term to be maximized. However, in an actual system, the distribution of the data arrival for each DU and channel state transition is often unavailable *a priori*, which makes it computationally impossible to directly optimize the long-term utility shown in (2). Similar to [21], we introduce an
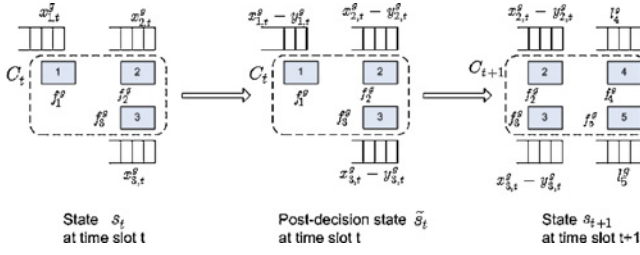
Fig. 2. Postdecision state illustration.

intermediate state which represents the state after transmitting the data (making the decision), but before the new data arrives and new channel state is realized. This intermediate state is referred to as the postdecision state $\tilde{s}_t$. In order to differentiate the "postdecision" state $\tilde{s}_t$ from the state s_t, we refer to the state s_t as the "normal" state. The postdecision state at time slot t is also illustrated in Fig. 2. From this figure, we know that the postdecision state is a deterministic function of the normal state s_t and the decision y_t, which is given by $\tilde{s}_t = (C_t, z_t, h_t)$, where $z_t = x_t - y_t$. In other words, after introducing the postdecision states, the decision y_t only affects the transition from the normal state s_t to the postdecision state $\tilde{s}_t$, and the dynamics only affect the transition from the postdecision state $\tilde{s}_t$ to the normal state s_{t+1}.

Similarly, we introduce the postdecision state-value function $U(C_t, z_t, h_t)$ to represent the optimal long-term utility starting from the postdecision state $\tilde{s}_t = (C_t, z_t, h_t)$. Then, we can rewrite the Bellman's equations in (4) as

$$U(C_t, z_t, h_t) = \mathbf{E}_{h_{t+1}|h_t, \mathbf{l}_{t+1}} V(C_{t+1}, z_t \oplus \mathbf{l}_{t+1}, h_{t+1}) \quad (5)$$

and

$$V(C_t, \mathbf{x}_t, h_t) = \max_{0 \leq y_t \leq x_t} \left\{ \sum_{f \in C_t} q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f \in C_t} y_{f,t} \right) + \alpha U(C_t, \mathbf{x}_t - \mathbf{y}_t, h_t) \right\}. \quad (6)$$

The first equation shows that the postdecision state-value function U is obtained from the normal state-value function V by taking the expectation over the possible data arrivals and possible channel transitions. The second equation shows that the normal state-value function is then obtained from the postdecision state-value function U by performing the maximization over the possible decision, which is referred to as the *foresighted decision* since the optimal decision is found by maximizing the long-term utility. However, when performing the foresighted decision illustrated in (6), we face the following challenges.

1) At each time slot, there are multiple DUs that are available for transmission. Determining the amount of data to transmit for each DU is a multivariable optimization which is often too complicated to solve at each time slot. However, it is fortunate that the DUs can be prioritized based on their heterogeneous data features. This prioritization will allow us to separate the multi-DU foresighted decision in (6) into multiple single-DU

foresighted decisions (which is single-variable optimization). The separation will be presented in Sections III-C.

2) In video transmission systems, we do not know the statistical knowledge of the underlying dynamics (e.g., channel state transition, the amount of packets for newly arriving DUs). However, after introducing the postdecision state $\tilde{s}_t = (C_t, z_t, h_t)$, we can separate the media transmission system into two phases: the foresighted decision phase, which is governed by (6) and the dynamic realization phase, which is governed by (5). We further notice that given the postdecision state-value function U, the foresighted decision phase is independent of the dynamic realization phase. This motivates us to directly learn the postdecision state-value function when the underlying dynamics are unknown. In Section III-D, we will present how the postdecision state-value functions can be learned over time for the separated foresighted decisions.

B. Transmission Priority of DUs

In this section, we aim to define the transmission priorities between DUs. At each time slot t, the optimal amount of data to be transmitted from DU $f \in C_t$ is denoted by $y_{f,t}^*$.

Definition 1 (Transmission Priority): At any time slot t, if $f, f' \in C_t$ and $(x_{f,t} - y_{f,t}^*) y_{f',t}^* = 0$ for any $x_t \geq 0$ and any channel state h_t, then DU f has a higher transmission priority than DU f', which is denoted by $f \triangleleft f'$.

The above definition on the priority indicates that when DU f has a higher transmission priority than DU f', then the data from DU f will be transmitted before the data from DU f' is transmitted. Given the optimal postdecision state value function $U(C_t, \mathbf{x}_t, h_t)$, we can prioritize the DUs as follows.

Lemma 1 (Prioritization Using Optimal Postdecision State Value Functions): For any two DUs $f, f' \in C_t$, if

$$U(C_t, \mathbf{x} + e_f, h_t) - U(C_t, \mathbf{x} + e_{f'}, h_t) < (q_f - q_{f'})/\alpha \quad \forall x \quad (7)$$

where e_f is a vector which has the same dimension as x and the element corresponding to DU $f \in C_t$ is 1 and the elements corresponding to all other DUs are 0, then $f \triangleleft f'$.

Proof: See Appendix A.

In Lemma 1, the left side of the inequality represents the long-term utility difference between the scenario that the DU f is not transmitted at the current time slot and the scenario that DU f' is not transmitted at the current time slot. The right side represents the distortion impact difference of two DUs (scaled by the discount factor). When the long-term utility difference is smaller than the distortion impact difference, then DU f has the higher transmission priority than DU f'.

However, as shown in Lemma 1, in order to determine the priorities of the DUs, we have to compute the optimal postdecision state value function first, which may not be possible in practical video transmission systems since we cannot obtain the postdecision state value function without first solving the Bellman's equations in (5) and (6). However, we are able to derive the priorities between the DUs based on the heterogeneous attributes of DUs without computing the optimal postdecision state value functions. ∎
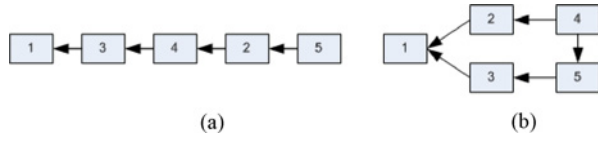
Fig. 3. Examples of priority graphs with five DUs. (a) All DUs are prioritized (chain). (b) Only part of the DUs are prioritized (e.g., the priorities of DUs 2 and 3 are unknown).

Lemma 2 (Prioritization Using the Heterogeneous Attributes of Independent DUs): For DUs $f, f' \in C_t$, if $q_f \geq q_{f'}$ and $d_f \geq d_{f'}$ (equalities do not hold at the same time), then $f \lhd f'$.

   Proof: See Appendix B.

The priority $f \lhd f'$ indicates that $\left(x_{f,t} - y_{f,t}^* \right) y_{f',t}^* = 0$ at any time slot t when $f, f' \in C_t$. This further implies that: 1) the buffer state $x_{f',t}$ of DU f' does not affect the decision on the amount of data, $y_{f,t}^*$, to be transmitted from DU f at any time slot t; and 2) when starting to transmit the data from DU f', all the data from DU f must be transmitted, i.e., the postdecision traffic state for DU f is zero. In the next section, we will utilize the priorities between DUs and present the separation in the multi-DU foresighted decision given in (6) and develop a low-complexity scheduling algorithm. ∎

C. Priority Graph-Assisted Scheduling

Given the transmission priority between DUs derived based on the DUs' attributes as shown in Lemma 2, we are able to construct a DAG to represent the priorities of the DUs at each time slot, which is referred to as the priority graph and denoted by $PG_t = \langle C_t, E_t \rangle$, where C_t is the set of DUs available for transmission and E_t is the set of edges representing the priorities between two DUs. In this priority graph, if $f \lhd f'$, then there is an edge in E_t pointed from DU f' to DU f. Two examples of priority graphs are shown in Fig. 3. Note that the priority graph is different from the dependence graph [7], which is built only based on the source coding dependences among DUs.

In the following, we will try to separate the multi-DU foresighted decision in (6) into multiple one-DU foresighted decisions. We first consider that the DUs available for transmission at each time slot can be fully prioritized [i.e., the corresponding priority graph is a chain as shown in Fig. 3 (a)]. The following theorem shows that we can decompose the multi-DU foresighted decision into multiple single-DU foresighted decisions at each time slot.

Theorem 1 (separation principle for multi-DU foresighted decision with priority graph of chain): When the DUs in each context are prioritized as a chain, then the optimal decision for each DU at each time slot can be computed as

$$y_{f,t}^* = \arg \max_{0 \leq y_{f,t} \leq x_{f,t}}$$
$$\left\{ q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f' \lhd f} x_{f',t} + y_{f,t} \right) \right. \tag{8}$$
$$\left. + \alpha U_f \left(C_t, x_{f,t} - y_{f,t}, h_t \right) \right\}$$

where the postdecision state value function $U_f \left(C_t, x, h_t \right)$ satisfies the following Bellman's equations:

if $f \in C_t \bigcap C_{t+1}$, then

$$U_f \left(C_t, x_{f,t}, h_t \right) = \underset{h_{t+1} | h_t}{\mathbf{E}} \underset{l_{f'} : f' \lhd f, f' \in C_{t+1} / C}{\mathbf{E}} \max_{0 \leq y_{f,t+1} \leq x}$$
$$\left\{ \begin{array}{c} q_f y_{f,t+1} - \lambda \rho \left(h_{t+1}, \sum_{\forall f' \lhd f, f' \in C_{t+1}/C_t} l_{f'} + y_{f,t+1} \right) + \\ \alpha U_f \left(C_{t+1}, x_{f,t} - y_{f,t+1}, h_{t+1} \right) \end{array} \right\} \tag{9}$$

and if $f \in C_t / C_{t+1}$, then $U_f \left(C_t, x, h_t \right) = 0$.

   Proof: See Appendix C. ∎

Remark 1: Theorem 1 indicates that we can find the optimal decision $y_{f,t}^*$ for each DU $f \in C_t$ by solving the foresighted decision given in (8) from the highest priority DU to the lowest priority DU. In this foresighted decision, the postdecision state-value function $U_f \left(C_t, x_{f,t} - y_{f,t}, h_t \right)$ only depends on the buffer state $x_{f,t} - y_{f,t}$ of DU f and is independent of the buffer states of the other DUs in the current context C_t. This is because when transmitting the data from DU f, the data from DU f' $(f' \lhd f)$ has been transmitted (i.e., the buffer state is zero), and the data from DU f' $(f \lhd f')$ cannot be transmitted [i.e., the buffer state will not affect the foresighted decision in (8)]. We note that when making the foresighted decision for DU f, the transmission cost is $\rho \left(h_t, \sum_{f' \lhd f} x_{f',t} + y_{f,t} \right) - \rho \left(h_t, \sum_{f' \lhd f} x_{f',t} \right)$, which is the marginal transmission cost of transmitting the data from DU f. However, the term $\rho \left(h_t, \sum_{f' \lhd f} x_{f',t} \right)$ is independent of the decision variable $y_{f,t}$ and hence, it is not shown in the foresighted decision in (8).

Remark 2: The postdecision state value function for DU f is computed as in (9). If DU f is expired at time slot $t + 1$, then the postdecision state value function is zero, otherwise it is computed by solving the Bellman's equations for DU f. In the Bellman's equations, we note that the postdecision state-value function for DU f is not affected by DUs f' $(f \lhd f')$. In fact, it only depends on the buffer states of the DUs that arrive at time slot $t + 1$ and have a higher priority than DU f. We can also note that the update of the postdecision state-value function for DU f' is not affected by DU f.

We now consider a general scenario where the priorities of DUs at each time slot are represented by a general priority graph instead of a chain. The priority graph for the DUs at time slot t is given by $PG_t = \langle C_t, E_t \rangle$. Similar to Theorem 1, for any two DUs $f, f' \in C_t$, if $f \lhd f'$, then we should transmit the data from DU f first and the buffer state of DU f' does not affect the foresighted decision for DU f. However, the buffer state of DU f will affect the transmission cost of DU f' in the foresighted decision, but will not affect the update of the postdecision state value function. If f and f' are not prioritized, we have to decide which DU should be transmitted first and how much data should be transmitted from this selected DU. The following theorem answers this question.

Theorem 2 (separation principle for multi-DU foresighted decision with general priority graph): Given the priority graph $PG_t = \langle C_t, E_t \rangle$ at time slot t, the optimal decisions for the DUs are performed as in Algorithm 1. After determining the optimal decisions, the postdecision state value function of DU f is updated as follows.

Algorithm 1: Optimal packet scheduling induced by the priority graph for independent DUs

Input: PG_t, $U_f\left(C_t, x_{f,t} - y_{f,t}, h_t\right)$

Initialize: $PG_t^0 = PG_t$

For $k = 1, \cdots, |C_t|$:

$$f^k = \arg \max_{f \in root\left(PG_t^k\right)} \max_{0 \le y_{f,t} \le x_{f,t}}$$

$$\left\{ q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{j=1}^{k-1} y_{fj,t}^* + y_{f,t} \right) + \alpha U_f \left(C_t, x_{f,t} - y_{f,t}, h_t \right) \right\} \quad (11)$$

$$y_{f^k,t}^* = \arg \max_{0 \le y_{f^k,t} \le x_{f^k,t}} \left\{ \begin{matrix} q_{f^k} y_{f^k,t} - \lambda \rho \left(h_t, \sum_{j=1}^{k-1} y_{fj,t}^* + y_{f^k,t} \right) \\ + \alpha U_{f^k} \left(C_t, x_{f^k,t} - y_{f^k,t}, h_t \right) \end{matrix} \right\} \quad (12)$$

where $root\left(PG_t^k\right)$ is the operator extracting the roots of the priority graph PG_t^k and $PG_t^k = PG_t^{k-1} / \left\{ f^{k-1} \right\}$.

Return $\left(f^1, \cdots, f^{|C_t|}\right)$ and $\left(y_{f^1,t}^*, \cdots, y_{f^{|C_t|},t}^*\right)$.

If $f \in C_t \cap C_{t+1}$, then

$$U_f\left(C_t, x_{f,t}, h_t\right) = \mathop{\mathbf{E}}_{h_{t+1}|h_t} \mathop{\mathbf{E}}_{l_{f'}: f' \tilde{\triangleleft} f, f' \in C_{t+1}/C} \max_{0 \le y_{f,t+1} \le x_{f,t}}$$

$$\left\{ \begin{matrix} q_f y_{f,t+1} - \lambda \rho \left(h_{t+1}, \sum_{f' \tilde{\triangleleft} f, f' \in C_{t+1}/C_t} l_{f'} + y_{f,t+1} \right) + \\ \alpha U_f \left(C_{t+1}, x_{f,t} - y_{f,t+1}, h_{t+1} \right) \end{matrix} \right\}$$

(10)

else, $U_f\left(C_t, x, h_t\right) = 0$, where $f' \tilde{\triangleleft} f$ means that the transmission order of DU f' is earlier than DU f, which is determined in Algorithm 1.

Proof: The proof can be derived similarly to that of Theorem 1.

The optimal packet scheduling illustrated in Algorithm 1 can be easily explained as follows. Starting from the priority graph PG_t, we compare the DUs that are the roots in the priority graph PG_t^k and select the DU with the highest long-term utility to transmit as shown in (11). The optimal scheduling for the selected DU is found by solving the corresponding foresighted decision as shown in (1). Finding the optimal packet scheduling in state $s_t = (C_t, \boldsymbol{x}_t, h_t)$ as illustrated in Algorithm 1 can also be interpreted by using a priority tree, which is uniquely constructed from the priority graph PG_t corresponding to the context C_t. Two examples of priority trees, which correspond to the priority graphs in Fig. 3, are given in Fig. 4. The root of the priority tree is the priority graph PG_t and each node is a priority graph. The child nodes of each node in the priority tree are obtained by removing one of the root packets in the priority graph at this node. Then, finding the optimal packet scheduling is equivalent to "traveling" the priority tree induced by the priority graph PG_t. ∎

The update of the postdecision state-value function $U_f\left(C_t, x, h_t\right)$ for DU f is performed independently of the other DUs in the current context C_t as shown in (10), which is the same as the update of the postdecision state value function for the fully prioritized DUs presented in Theorem 1. However, unlike in Theorem 1, when updating the postdecision state-value function $U_f\left(C_t, x, h_t\right)$, we cannot directly prioritize all the DUs arriving at time slot $t + 1$, i.e., the DUs in the set of $C_{t+1}\backslash C_t$ because we may not be able to compare the transmission priority between the arriving DUs with the DUs persisting from time slot t, i.e., in the set of $C_{t+1} \cap C_t$. Hence,
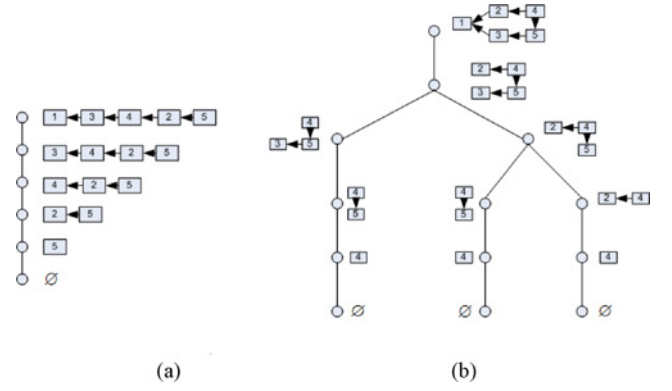

Fig. 4. Priority trees induced from (a) graph in which all packets are prioritized (chain), and (b) graph in which some packets are prioritized and others are not.

we resort to the priority graph built for the DUs in the set of $C_{t+1}\backslash C_t \bigcup \{f\}$ and Algorithm 1 in order to determine the transmission orders of the DUs in this set. When the transmission order of the DUs $C_{t+1}\backslash C_t \bigcup \{f\}$ is determined, we can update the postdecision state-value function $U_f\left(C_t, x, h_t\right)$ for DU f which only depends on the DUs that are transmitted before it (i.e., DUs f' such that $f' \tilde{\triangleleft} f$) at the same time slot. It is easy to show that Algorithm 1 preserves the transmission priority (i.e., DUs with higher priorities are always transmitted before DUs with lower priorities).

D. Online Learning

In multimedia transmission systems, we do not know the statistical knowledge about the packet arrivals for each DU and the underlying channel state transition. Without this statistical knowledge, we cannot directly update the postdecision state value function for each DU as shown in (10). In the following, we show how we can update the postdecision state-value functions for the DUs without having the statistical knowledge about the underlying dynamics.

From (10), we know that the expectation over the dynamics is performed outside of the maximization and that the dynamics (including the packet arrivals and channel state transition) are independent of the buffer states. Then, we are able to update the postdecision state-value function using time-average as presented in [22]. That is, we can update the postdecision state-value function for all the possible buffer states for each DU at each time slot. The online learning algorithm for each DU $f \in C_t$ is presented as follows. At time slot t, we first perform Algorithm 1 to determine the optimal decision for all the DUs in the context of C_t and the transmission order $\left(f^1, \cdots, f^{|C_t|}\right)$. Then the postdecision state-value function for DU $f \in C_t \bigcap C_{t-1}$ is updated for all the possible buffer state $x \in \left[0, l_f^{\max}\right]$ as follows.

Online learning for independent DUs

$$U_{f,t}\left(C_{t-1}, x, h_{t-1}\right) =$$

$$(1-\beta_t) \max_{0 \le y_{f,t+1} \le x} \left\{ \begin{matrix} q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f' \tilde{\triangleleft} f, f' \in C_t/C_{t-1}} l_{f'} + y_{f,t} \right) \\ + \alpha U_{f,t-1} \left(C_t, x - y_{f,t}, h_t \right) \end{matrix} \right\}$$

$$+ \beta_t U_{f,t-1}\left(C_{t-1}, x, h_{t-1}\right)$$

(13)

where β_t is a diminishing step size, e.g., $\beta_t = 1/t$.

We highlight the key ideas of the convergence proof of the online learning algorithm described above and refer the interested reader to [22] for more details of the proof.

First, in Theorems 1 and 2, we can separate the postdecision state-value function $U(C_t, \boldsymbol{x}_t, h_t)$ into multiple postdecision state-value functions $\{U_f(C_t, x_{f,t}, h_t), f \in C_t\}$, each corresponding to one DU in the current context. Then each postdecision state-value function $U_f(C_t, x_{f,t}, h_t)$ is updated subsequently based on the transmission priority. The advantage of this separation is that it allows us to update each 1-D postdecision state-value function $U_f(C_t, x_{f,t}, h_t)$ subsequently, given the current context C_t and channel state h_t instead of the original multidimensional function $U(C_t, \boldsymbol{x}_t, h_t)$.

Second, our online learning algorithm transmits the data according to the transmission priority as shown in Algorithm 1. From Lemma 1, we know that transmitting the DUs with higher transmission priority increases the postdecision state-value function $U_f(C_t, x_{f,t}, h_t)$. Hence, it can be easily shown that $U_f(C_t, x_{f,t}, h_t)$ is a concave function of $x_{f,t}$ given the current context C_t and channel state h_t.

Third, in [22], we show that the concave function $U_f(C_t, x_{f,t}, h_t)$ can be approximated by a piecewise linear function and updated on-the-fly, which can significantly reduce the complexity of updating the postdecision state value function.

Fourth, the on-the-fly updating of the postdecision state-value function will converge since the updating procedure in (10) is a maximum norm α-contractor.

IV. PACKET SCHEDULING FOR INTERDEPENDENT DUS

In this section, we aim to develop a packet scheduling solution for interdependent DUs. Different from the independent DUs case, due to interdependences, the scheduling decision on each DU f will be affected by the amount of data transmitted from the DUs on which DU f depends. As discussed in Section II, the dependence between DUs is expressed by a DAG which is different from the priority graph defined in Section III-B. In order to capture the impact of the dependence between the DUs, we introduce a dependence factor $p_f(v) \in [0, 1]$ for each DU f to represent the impact on the distortion reduction of those DUs that depend on DU f, which is a function of the amount of data remaining when DU f is expired. One example of dependence factor is $p_f(z) = \exp(-\beta_f z)$ as given in [23] and [24]. Then, the utility at each time slot is given by

$$u(s_t, \boldsymbol{y}_t) = \sum_{f \in C_t} \prod_{f' \prec f} p_{f'}(z_{f',d_{f'}}) q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f \in C_t} y_{f,t}\right) \tag{14}$$

where $z_{f',d_{f'}}$ is the remaining amount of data in the postdecision state of DU f' at time slot $d_{f'}$ (i.e., the amount of data from DU f' that are not received by the decoder). The difference between (14) and (1) is that the distortion reduction in (14) depends on not only the number of packets transmitted for each DU as shown in (1), but also the dependence factors

of the DUs that the current DUs depend on. The long-term utility is the same as in (2).

For the interdependent DUs, in order to capture the Markovian property of the scheduling problem, we define the state at each time slot t as $s_t = (C_t, p_t, \boldsymbol{x}_t, h_t)$ to include the current context C_t, buffer states $\boldsymbol{x}_t$, channel state h_t and dependence factor vector $\boldsymbol{p}_t$ from the parent DUs. The dependence factor vector $\boldsymbol{p}_t$ is given by $\boldsymbol{p}_t = \left[p_{f'}(z_{f',d_{f'}})\right]_{f' \prec f, f \in C_t}$. That is, the dependence factor vector includes all the dependence factors of the DUs that the DUs in the set C_t depend on. The postdecision state is defined as the state after the scheduling decision but before the new DU arrivals and the new channel state realization. We note that the postdecision dependence factor vector is the same as $\boldsymbol{p}_t$. We directly use $\boldsymbol{p}_t$ as the postdecision dependence factor vector in the postdecision state. Hence, the multi-DU foresighted decision based on the postdecision states is given as follows:

$$V(C_t, \boldsymbol{p}_t, \boldsymbol{x}_t, h_t) = \max_{0 \leq \boldsymbol{y}_t \leq \boldsymbol{x}_t}$$
$$\left\{ \begin{array}{l} \sum_{f \in C_t} \prod_{f' \prec f} p_{f'}(z_{f',d_{f'}}) q_f y_{f,t} - \lambda \rho\left(h_t, \sum_{f \in C_t} y_{f,t}\right) \\ + \alpha U(C_t, \boldsymbol{p}_t, \boldsymbol{x}_t - \boldsymbol{y}_t, h_t) \end{array} \right\} \tag{15}$$

where $U(C_t, \boldsymbol{p}_t, \boldsymbol{z}_t, h_t)$ is the postdecision state value function.

Similar to the independent DUs, we aim to separate the multi-DU foresighted decision in (15) into multiple single-DU foresighted decision. We can introduce the priority between interdependent DUs to differentiate the transmission orders of the DUs in the current context C_t. However, due to the dependence, we cannot directly apply Lemma 2 here. Instead, we can prioritize the DUs using their heterogeneous attributes in the following lemma.

Lemma 1 (prioritization using heterogeneous attributes of): interdependent DUs For any context C_t, if $f, f' \in C_t$ and $f \prec f'$, then $f \triangleleft f'$.

Proof: First, we notice that if $f \prec f'$, then $q_f \geq q_{f'}$ and $d_f \leq d_{f'}$. Furthermore, when $f \prec f'$, from (14), we know that the gained distortion reduction from DU f' (i.e., $p_{f',t} q_{f'} y_{f',t}$) is discounted by the dependence factor $p_{f',t}$ which is impacted by the amount of remaining data at DU f. In other words, transmitting the data from DU f will always achieve higher long-term utility than transmitting the data from DU f', which means $f \triangleleft f'$. ∎

From Lemma 3, we note that, based on the dependences among DUs, we can construct the priority graph $PG_t = \langle C_t, E_t \rangle$ for each context C_t. It is clear that the priority graph PG_t is the dependence graph corresponding to the DUs in the current context C_t. At time slot t, given the priority graph PG_t and the dependence factors vector $\boldsymbol{p}_{f,t} = \left[p_{f'}(z_{f',d_{f'}})\right]_{f' \prec f}$ for each DU $f \in C_t$, we can perform the foresighted decision for each DU as in Algorithm 2. In Algorithm 2, we separate the multi-DU foresighted decision and perform it by traveling the priority tree as illustrated in Fig. 4, which preserves the priorities between DUs.

Similarly, we can also update the postdecision state value function $U_f(C_t, \boldsymbol{p}_{f,t}, x_{f,t}, h_t)$ as follows.

Algorithm 2: Optimal Packet Scheduling Induced by the Priority Graph for Interdependent DUs

---

Input: PG_t, $\boldsymbol{p}_t$, $U_f\left(C_t, \boldsymbol{p}_{f,t}, x_{f,t}, h_t\right)$

Initialize: $PG_t^0 = PG_t$

For $k = 1, \cdots, |C_t|$:

$f^k =$

$$\arg\max_{f \in root\left(PG_t^k\right)} \max_{0 \leq y_{f,t} \leq x_{f,t}} \left\{ \begin{array}{l} \mathbf{1}^T \boldsymbol{p}_{f,t} \cdot q_f y_{f,t} - \lambda\rho\left(h_t, \sum_{j=1}^{k-1} y_{f^j,t}^* + y_{f,t}\right) \\ +\alpha U_f\left(C_t, p_{f,t}, x_{f,t} - y_{f,t}, h_t\right) \end{array} \right\} \quad (16)$$

$y_{f^k,t}^* =$

$$\arg\max_{0 \leq y_{f^k,t} \leq x_{f^k,t}} \left\{ \begin{array}{l} \mathbf{1}^T \boldsymbol{p}_{f^k,t} \cdot q_{f^k} y_{f^k,t} - \lambda\rho\left(h_t, \sum_{j=1}^{k-1} y_{f^j,t}^* + y_{f^k,t}\right) \\ +\alpha U_{f^k}\left(C_{f^k}, \boldsymbol{p}_{f^k,t}, x_{f^k,t} - y_{f^k,t}, h_t\right) \end{array} \right\} \quad (17)$$

where $root\left(PG_t^k\right)$ is the operator extracting the roots of the priority graph PG_t^k, $PG_t^k = PG_t^{k-1}/\left\{f^{k-1}\right\}$ and update $p_{f,t}$, $f \neq f^1, \cdots, f^k$.

Return $\left(f^1, \cdots, f^{|C_t|}\right)$ and $\left(y_{f^1,t}^*, \cdots, y_{f^{|C_t|},t}^*\right)$.

Online learning for interdependent DUs: If $f \in C_{t-1} \bigcap C_t$ (i.e., DU f is not expired at both time slots)

$$U_{f,t}\left(C_{t-1}, \boldsymbol{p}_{f,t-1}, x, h_{t-1}\right) =$$

$$(1 - \beta_t) \max_{0 \leq y_{f,t+1} \leq x} \left\{ \begin{array}{l} \mathbf{1}^T \boldsymbol{p}_{f,t} \cdot q_f y_{f,t} - \lambda\rho\left(h_t, \sum_{\substack{f^i, i < k \\ f^i \in C_t/C_{t-1}}} l_{fi} + y_{f,t}\right) \\ +\alpha U_{f,t-1}\left(C_t, \boldsymbol{p}_{f,t-1}, x - y_{f,t}, h_t\right) \end{array} \right\} \cdot$$

$$+\beta_t U_{f,t-1}\left(C_{t-1}, \boldsymbol{p}_{f,t-1} x, h_{t-1}\right)$$

$$\quad (18)$$

If $f \in C_{t-1}\backslash C_t$ (i.e., DU f is expired at time slot t)

$$U_{f,t}\left(C_{t-1}, \boldsymbol{p}_{f,t-1}, x, h_{t-1}\right) = (1 - \beta_t) \sum_{f': f \prec f'} \max_{0 \leq y_{f',t+1} \leq x}$$

$$\left\{ \begin{array}{l} \mathbf{1}^T \boldsymbol{p}_{f',t-1} \cdot q_{f'} y_{f',t} - \lambda\rho\left(h_t, \sum_{\substack{f^i, i < k \\ f^i \in C_t/C_{t-1}}} l_{fi} + y_{f',t}\right) \\ +\alpha U_{f',t-1}\left(C_t, \boldsymbol{p}_{f',t-1}, x - y_{f',t}, h_t\right) \end{array} \right\} \cdot$$

$$+\beta_t U_{f,t-1}\left(C_{t-1}, \boldsymbol{p}_{f,t-1}, x, h_{t-1}\right)$$

$$\quad (19)$$

When DU f is not expired at both time slots $t-1$ and t, we update the postdecision state value function using a time-average similar to the one in (13). However, when DU f is expired at time slot t (i.e., $f \in C_{t-1}\backslash C_t$), due to the dependence, the postdecision state of DU f at time slot $t-1$ will affect the decision of those DUs in the context C_t that depend on DU f. Hence, the postdecision state value function of DU f is updated as in (19) to take into account the dependence impact on the descendent DUs, which is different from the case of independent DUs, where the postdecision state value function is zero.

However, since the dependence-factor vector $\boldsymbol{p}_{f,t}$ often has high dimensionality (because the DUs in the set C_t depend on multiple DUs) and takes real values in the range of $[0, 1]$, it is difficult to compute and store the postdecision state value function directly. Instead of computing the post-decision state-value function $U_f\left(C_t, \boldsymbol{p}_{f,t}, z_t, h_t\right)$ directly, we approximate it by $\mathbf{1}^T \boldsymbol{p}_{f,t} \cdot U_f\left(C_t, z_t, h_t\right)$, where $U_f\left(C_t, z_t, h_t\right)$ is the postdecision state-value function corresponding to the

Channel Gain (h^2/σ^2) Regions	Representative States
(0, 0.0280], (0.0280, 0.0580] (0.0580, 0.0960] (0.0960, 0.1400] (0.1400, 0.1980] (0.1980, 0.2780], (0.2780, 0.4160] (0.4160, ∞]	0.0131, 0.0418, 0.0753, 0.1157, 0.1661, 0.2343, 0.3407, 0.6200

case that $\boldsymbol{p}_{f,t} = \mathbf{1}$ which means that all the DUs that DU f depends on are successfully received. Then $U_f\left(C_t, z_t, h_t\right)$ can be updated using (18) and (19) by setting $\boldsymbol{p}_{f,t} = \mathbf{1}$ and $\boldsymbol{p}_{f',t} = \mathbf{1}, \forall f \prec f'$. It is clear that the approximation allows us to represent the postdecision state-value function as presented in Section III-D, which significantly reduces the dimensionality of the postdecision state-value function. This is because the dependence-factor vector is not the component of the arguments in the approximated postdecision state-value function any longer.

V. SIMULATION RESULTS

In this section, we perform several numerical experiments to verify the performance of the proposed framework and compare with various state-of-the-art solutions for multimedia communications.

A. Performance Comparison of Various Packet Scheduling Solutions for Video Transmission

In this section, we compare our proposed packet scheduling solution with start-of art solutions which only consider either the heterogeneous media characteristics or the time-varying channel conditions.

The energy function for transmitting the amount of y (in packets) traffic at the channel state h is given by $c(h, y) = \sigma^2\left(2^{Ly} - 1\right)/|h|^2$, where σ^2 is the variance of the white Gaussian noise [20]. In this simulation, we choose $\bar{h}^2/\sigma^2 = 0.14$, where $\bar{h}$ is the average channel gain. We divide the entire channel gain range into eight regions each of which is represented by a representative state. The states are presented in Table II. We choose $\alpha = 0.95$. The transmission system is time-slotted with the time slot length of 10 ms.

We consider three comparable solutions: 1) our proposed packet scheduling solution which takes into account both the heterogeneous multimedia traffic characteristics (e.g., delay deadlines, distortion impacts and dependences, and so on) and time-varying network conditions; 2) the packet scheduling solution [6] which only considers the distortion impact of each media packet and the observed channel conditions and is referred to as "distortion-impact" driven packet scheduling; and 3) the packet scheduling solution obtained by solving the rate-distortion optimization assuming the constant channel conditions (i.e., using average channel conditions) and linear transmission cost as in RaDiO [7], [12], which is referred to as the rate-distortion optimized packet scheduling.

In the first experiment, to compress the video data, we used a wavelet-based scalable video coding scheme [15], which

is attractive for wireless streaming applications because it provides on-the-fly application adaptation to channel conditions, support for a variety of wireless receivers with different resource capabilities and power constraints, and easy prioritization of various coding layers and video packets. We choose for this experiment three video sequences (*Foreman*, *Coastguard* and *Mobile* at CIF resolutions, 30 frames/s), exhibiting different motion activities. The video sequences *Foreman* and *Coastguard* are encoded at the bit rate of 512 kb/s and *Mobile*, due to its high-frequent texture and complicated motion, is encoded at 1024 kb/s. In this simulation, each GOP contains 16 frames and each encoded video frame can tolerate a delay of 266 ms, corresponding to the half duration of a GOP. The packet arrival distribution $PMF_{f_j^g}(l)$ of each DU is content-dependent and is not required to know in our simulation. The scheduling time window W is set to 266 ms. The tradeoff parameter λ is varied from 1 to 30 to get different quality-energy tradeoffs.

In Fig. 5(a)–(c), we show the peak-signal-to-noise ratio (PSNR) as a function of the consumed energy curves under the different scheduling solutions for the three video sequences. From these figures, we note that our proposed cross-layer optimization solution outperforms both the "distortion-impact" driven packet scheduling and rate-distortion optimized packet scheduling by, on average, around 2 dB and 5 dB in *Foreman*, 1.5 dB and 3.5 dB in *Coastguard*, and 0.5 dB and 2.5 dB in *Mobile* in terms of PSNR. The improvement comes from the fact that, in our proposed solution, as shown in (15), we schedule the video packets at each time slot based on not only the current packet distortion, the estimated dependence factor from the previous video packets, the incurred transmission cost, but also the impact of the current video packets on the following video packets through postdecision state-value function. Hence, the heterogeneous characteristics of the multimedia packets as well as the time-varying channel conditions are automatically taken into account when we make a scheduling decision. In contrast to our solution, the rate-distortion optimized packet scheduling solution only considers the current packet distortion, the dependence factor from the previous packets and the impact on the following video packets through the dependence factor including both the previous and current packets. However, it does not consider the incurred transmission cost and time-varying channel conditions. The "distortion-impact"-driven solution only considers the current packet distortion and time-varying channel conditions. We also notice that the "distortion-impact"-driven solution obtains higher received video quality than the rate-distortion optimized packet scheduling. It shows that the time-varying channel conditions and the characteristics (dependences, distortion impacts and delay constants) of media packets play a very important role in improving the media quality.

In the second experiment, we compress the video sequence (*CrowdRun* at resolution 720×540, 30 frames/s) using H.264 SVC [17]. The video sequence is encoded at 2 Mb/s with the GOP structure of "IBBBBBBBP" and each frame having three quality layers. In this simulation, each GOP contains eight frames and each encoded video frame can tolerate a delay of 266 ms, corresponding to the duration of a GOP. We transmit
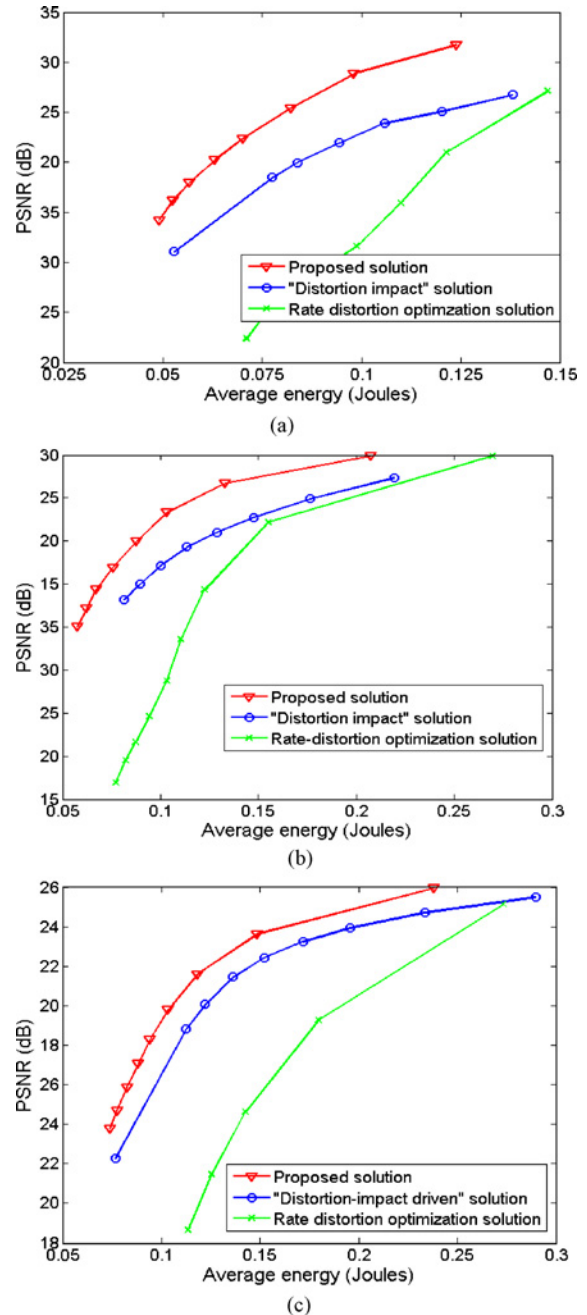

Fig. 5. PSNR-energy curve of (a) *Foreman*, (b) *Coastguard*, and (c) *Mobile* sequences for different transmission solutions.

the encoded video using three scheduling methods mentioned above under the same energy constraint (0.5 Joule). We get the video quality (in terms of PSNR) of 38.0 dB, 36.9 dB, and 36.1 dB, for the proposed solution, "distortion impact"-driven solution and RaDiO-based solution, separately. The quality improvement comes from the fact that our proposed solution is able to explore both the traffic heterogeneity and channel variations. This experiment also illustrates that the proposed scheduling solution is applicable to different video codecs [16], [17] once the dependences of video packets can be modeled as direct-acyclic graphs and the distortion impact of each packet can be computed.
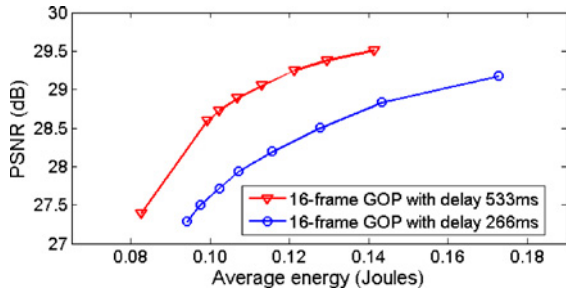
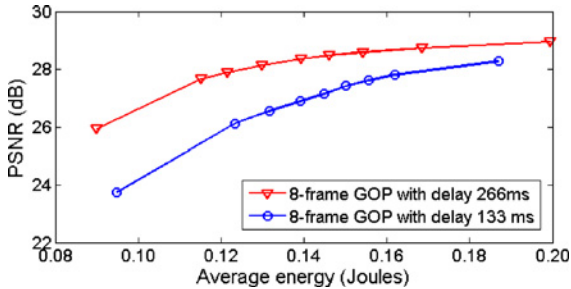Fig. 6. Video quality of *Coastguard* sequence with various delay deadlines when 16-frame GOP is used.

Fig. 7. Video quality of *Coastguard* sequence with various delay deadlines when 8-frame GOP is used.

B. Performance of Packet Scheduling Optimization with Various Delay Constraints and GOP Structures

In this section, we further compare the performance of the packet scheduling optimization solutions for streaming the *Coastguard* video sequence with various delay constraints and GOP structures. The wireless channel settings are the same as in Section V-A. We compare our solution with different combinations of delay deadlines and GOP structures. The PSNR versus consumed energy curves are given in Figs. 6 and 7. From Fig. 6, we notice that, when the video sequence is encoded with the GOP of 16 frames, by increasing the delay from 266 ms to 533 ms, the packet scheduling optimization can improve, on average, 1 dB in terms of PSNR. From Fig. 7, we further notice that, when the video sequence is encoded with the GOP of 8 frames, by increasing the delay from 133 ms to 266 ms, the packet scheduling optimization can improve, on average, 1.5 dB in terms of PSNR. The improvement comes from the fact that by increasing the delay, each media packet has more transmission opportunities and will be scheduled for transmission when a better channel condition is encountered. By increasing the number of frames in one GOP, the video sequence can be encoded more efficiently and there are fewer packets to be transmitted, which accordingly improves the video quality.

VI. CONCLUSION

In this paper, we formulated the problem of packet scheduling optimization for delay-sensitive packetized media applications as a Markov decision process. Based on the heterogeneous characteristics of the media packets, we expressed the transmission priorities between DUs as a DAG. Using the DAG expression, we were able to separate the multidata unit foresighted decision at each time slot into multiple

single-data unit foresighted decisions, which can subsequently be performed from the high priority DU to the low priority DU. The postdecision state-value function associated with each DU is updated individually using the online learning algorithms. The simulation results show that the proposed foresighted optimization solution significantly outperforms the start-of-art solutions which (partially) ignore the media characteristics and time-varying network conditions.

It is worth noting that error concealment employed at the decoder side may introduce different interdependences among DUs. For example, when a direct copy of the collocated data is used for error concealment, the dependences between the collocated DUs should be introduced even though there is no prediction between them. Indeed, our framework can be augmented to allow for different error concealment strategies by redefining the distortion impact and interdependences among DU, which will be an interesting area of future research. Moreover, note that this proposed systematic scheduling framework is general and can be easily applied to many other multimedia-related problems. For example, using the proposed context to represent the heterogeneous video data encoded by different video codecs, e.g., H.264 [16], SVC [15], [17], and so on, the proposed packet scheduling can also be applied to energy-efficient video encoding/decoding systems with dynamic voltage scaling, by separating the multi-DU scheduling decision into multiple single-DU scheduling decisions. When the transmission acknowledgement is delayed, as it is the case for multihop wired and wireless networks, we can easily extend the proposed scheduling framework using a partially-observed MDP formulation and then apply the proposed separation for the foresighted decision. We further noticed that the packet scheduling algorithm can also be deployed in a multihop network (e.g., mesh or sensor networks) to relay multimedia data from other nodes.

APPENDIX A
PROOF OF LEMMA 1

Proof: The optimal decision at time slot t is denoted by $\boldsymbol{y}_t^* = \left(y_{f,t}^*, y_{f',t}^*, \boldsymbol{y}_{-f-f',t}^*\right)$ where $\boldsymbol{y}_{-f-f',t}^*$ represents the optimal decision for the DUs other than DUs f, f'. We assume that $\left(x_{f,t} - y_{f,t}^*\right) y_{f',t}^* \neq 0$, which means that $x_{f,t} - y_{f,t}^* > 0$ and $y_{f',t}^* > 0$. We consider another decision $\boldsymbol{y}_t = \left(y_{f,t}^* + 1, y_{f',t}^* - 1, \boldsymbol{y}_{-f-f',t}^*\right)$ which is feasible since $x_{f,t} - y_{f,t}^* > 0$ and $y_{f',t}^* > 0$. We compare the long-term utility associated with the decision $\boldsymbol{y}_t$ to the one associated with the optimal decision $\boldsymbol{y}_t^*$ (i.e., $V\left(C_t, \boldsymbol{x}_t, h_t\right)$) as follows:

$$-q_f \left(x_{f,t} - y_{f,t}^* - 1\right) - q_{f'} \left(x_{f',t} - y_{f',t}^* + 1\right) -$$
$$\sum_{f'' \in C_t \backslash C_{t+1}} q_{f''} \left(x_{f'',t} - y_{f'',t}^*\right) - \lambda \rho \left(h_t, \sum_{f'' \in C_t} y_{f'',t}\right)$$
$$+ \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}_t^* - e_f + e_{f'}, h_t\right)$$
$$= q_f - q_{f'} + \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}_t^* - e_f + e_{f'}, h_t\right) -$$
$$\alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}_t^*, h_t\right) + V\left(C_t, \boldsymbol{x}_t, h_t\right) > V\left(C_t, \boldsymbol{x}_t, h_t\right).$$

The inequality is due to the fact that

$$q_f - q_{f'} + \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}_t^* - e_f + e_{f'}, h_t\right) - \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}_t^*, h_t\right)$$
$$= q_f - q_{f'} + \alpha U\left(C_t, \boldsymbol{x}' - \boldsymbol{y}_t^* + e_{f'}, h_t\right)$$
$$- \alpha U(C_t, \boldsymbol{x}' - \boldsymbol{y}_t^* + e_f h_t) > 0$$

where $\boldsymbol{x}' = \boldsymbol{x}' - e_f$ and the inequality is from the condition given in (7).

Hence, $\left(x_{f,t} - y^*_{f,t}\right) y^*_{f',t} \neq 0$ implies that $\boldsymbol{y}^*_t$ is not the optimal decision which contradicts the assumption.   ∎

APPENDIX B
PROOF OF LEMMA 2

Proof: To prove this, we only need to show that

$$U(C_t, \boldsymbol{x} + e_f, h_t) - U(C_t, \boldsymbol{x} + e_{f'}, h_t) < (q_f - q_{f'})/\alpha \,\forall \boldsymbol{x},\, t \le d_f.$$

We prove this using backward induction.

At time slot $t = d_f$, on the one hand, we have $U\left(C_t, \boldsymbol{x} + e_f, h_t\right) = U\left(C_t, \boldsymbol{x}, h_t\right)$ because DU f will expire and be deleted at the next slot and have no contribution to the postdecision state value function. On the other hand, we can prove that $0 \le U\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) - U\left(C_t, \boldsymbol{x}, h_t\right) \le q_{f'}$ because the best utility we can obtain in the future time slots by transmitting one packet in DU f' is $q_{f'}$. Then

$$-q_{f'} \le U\left(C_t, \boldsymbol{x} + e_f, h_t\right) - U\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right)$$

$$= U\left(C_t, \boldsymbol{x}, h_t\right) - U\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) \le 0 < \left(q_f - q_{f'}\right)/\alpha.$$

Now, we assume that at time slot $t \le d_f$, $U\left(C_t, \boldsymbol{x} + e_f, h_t\right) - U\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) < \left(q_f - q_{f'}\right)/\alpha, \forall \boldsymbol{x}$. We try to prove that $U\left(C_{t-1}, \boldsymbol{x} + e_f, h_{t-1}\right) - U\left(C_{t-1}, \boldsymbol{x} + e_{f'}, h_{t-1}\right) < \left(q_f - q_{f'}\right)/\alpha, \forall \boldsymbol{x}$ at time slot t. From the Bellman's equation in (5), we know that $U\left(C_{t-1}, \boldsymbol{x}_{t-1} + e_f, h_{t-1}\right) = \mathbf{E}V\left(C_t, \boldsymbol{x}_t + e_f, h_t\right)$ where $\boldsymbol{x}_{t+1}$ is derived from $\boldsymbol{x}_t$ by deleting the expired DUs and adding the new arriving DUs. Hence, it is equivalent to proving that $V\left(C_t, \boldsymbol{x} + e_f, h_t\right) - V\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) < \left(q_f - q_{f'}\right)/\alpha, \forall \boldsymbol{x}$. We denote the optimal scheduling decision in computing $V\left(C_t, \boldsymbol{x}, h_t\right)$ by $\boldsymbol{y}^*_t(\boldsymbol{x})$. Then the optimal decision in computing $V\left(C_t, \boldsymbol{x} + e_f, h_t\right)$ can be in three cases: 1) $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_f\right) = \boldsymbol{y}^*_t(\boldsymbol{x})$, i.e., the additional packet in DU f is not transmitted; 2) $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_f\right) = \boldsymbol{y}^*_t(\boldsymbol{x}) + e_f - e_{f''}$, i.e., transmitting the additional packet in DU f instead of the packet in DU f''; and 3) $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_f\right) = \boldsymbol{y}^*_t(\boldsymbol{x}) + e_f$, i..e transmitting the additional packet in DU f. Similarly, the optimal decision in computing $V\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right)$ also has three cases. However, we have the following relationship: if $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_f\right)$ is case $i = 1, 2, 3$, then $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_{f'}\right)$ should be case $i' = 1, \cdots, i$. For all the cases, we can prove that $V\left(C_{t+1}, \boldsymbol{x}_{t+1} + e_f, h_{t+1}\right) - V\left(C_{t+1}, \boldsymbol{x}_{t+1} + e_{f'}, h_{t+1}\right) \le q_f - q_{f'}/\alpha$. For example, we consider that $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_f\right)$ is case 3 and $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_{f'}\right)$ is case 1. Then

$$V\left(C_t, \boldsymbol{x} + e_f, h_t\right) = \sum_{f'' \in C_t} q_{f''} y^*_{f'',t} + q_f -$$
$$\lambda \rho \left(\sum_{f'' \in C_t} q_{f''} y^*_{f'',t} + 1, h_t\right) + \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}^*_t(\boldsymbol{x}), h_t\right)$$ and
$$V\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) =$$
$$\sum_{f'' \in C_t} q_{f''} y^*_{f'',t} - \lambda \rho \left(\sum_{f'' \in C_t} q_{f''} y^*_{f'',t}, h_t\right)$$
$$+ \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}^*_t(\boldsymbol{x}) + e_{f'}, h_t\right).$$

It is clear that

$$V\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) =$$
$$\sum_{f'' \in C_t} q_{f''} y^*_{f'',t} - \lambda \rho \left(\sum_{f'' \in C_t} q_{f''} y^*_{f'',t}, h_t\right)$$
$$+ \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}^*_t(\boldsymbol{x}) + e_{f'}, h_t\right)$$
$$\ge \sum_{f'' \in C_t} q_{f''} y^*_{f'',t} + q_{f'} - \lambda \rho \left(\sum_{f'' \in C_t} q_{f''} y^*_{f'',t} + 1, h_t\right)$$
$$+ \alpha U\left(C_t, \boldsymbol{x} - \boldsymbol{y}^*_t(\boldsymbol{x}), h_t\right)$$

since $\boldsymbol{y}^*_t\left(\boldsymbol{x} + e_{f'}\right)$ is in case 3. Hence, we can have $V\left(C_t, \boldsymbol{x} + e_f, h_t\right) - V\left(C_t, \boldsymbol{x} + e_{f'}, h_t\right) \le q_f - q_{f'}$. We can similarly prove the other scenarios. Finally, we prove that $U\left(C_{t-1}, \boldsymbol{x} + e_f, h_{t-1}\right) - U\left(C_{t-1}, \boldsymbol{x} + e_{f'}, h_{t-1}\right) < \left(q_f - q_{f'}\right)/\alpha$.   ∎

APPENDIX C
PROOF OF THEOREM 1

Proof: Since the DUs at each time slot are prioritized as a chain, from Lemma 1, we know that the optimal decision for each DU is found, starting from the highest priority DU, by solving the following optimization:

$$y^*_{f,t} = \arg \max_{0 \le y_{f,t} \le x_{f,t}} \left\{ \begin{array}{l} q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f' \lhd f, f' \in C_t} x_{f',t} + y_{f,t}\right) + \\ \alpha U\left(C_t, \{0_{f',t}\}_{f' \lhd f} \bigcup \{x_{f,t} - y_{f,t}\} \right. \\ \left. \bigcup \{x_{f',t}\}_{f \lhd f'}, h_t\right) \end{array} \right\}$$

where $0_{f',t}$ represents that DU f' is empty. As we know, when performing the foresighted decision for DU f, all the data from DU f' ($f' \lhd f$) has been transmitted and no data from DU f' ($f \lhd f'$) is transmitted. It is true for any time slot. Hence, we are able to split the postdecision state-value function in the above foresighted optimization into two parts

$$U\left(C_t, \{0_{f',t}\}_{f' \lhd f} \bigcup \{x_{f,t} - y_{f,t}\} \bigcup \{x_{f',t}\}_{f \lhd f'}, h_t\right) =$$
$$U_f\left(\left(C_t, \{x_{f,t} - y_{f,t}\}, h_t\right)\right) + U_{\{f \lhd f'\}}\left(C_t, \{x_{f',t}\}_{f \lhd f'}, h_t\right)$$

where $U_f\left(\left(C_t, \{x_{f,t} - y_{f,t}\}, h_t\right)\right)$ represents the long-term utility obtained for DU f and $U_{\{f \lhd f'\}}\left(C_t, \{x_{f',t}\}_{f \lhd f'}, h_t\right)$ represents the long-term utility obtained for all the DUs f' ($f \lhd f'$). The reason that we can split it is that the lower priority DU f' ($f \lhd f'$) will not affect the foresighted decision for DU f and the data from the lower priority DU can be transmitted only if the DUs from the higher priority DUs have been transmitted (i.e., empty) at the current time slot. Hence, $U_{\{f \lhd f'\}}\left(C_t, \{x_{f',t}\}_{f \lhd f'}, h_t\right)$ only depends on the amount of data from DU f ($f \lhd f'$) observed before the foresighted decision (i.e., $x_{f,t}$) and is independent of the decision at current time slot. Hence, the foresighted decision for DU f can be

rewritten as

$$y_{f,t}^* = \arg \max_{0 \leq y_{f,t} \leq x_{f,t}}$$
$$\left\{ q_f y_{f,t} - \lambda \rho \left(h_t, \sum_{f' \lhd f, f' \in C_t} x_{f',t} + y_{f,t} \right) + \alpha U_f \right.$$
$$\left. \left(C_t, x_{f,t} - y_{f,t}, h_t \right) \right\}$$

which is the form given in (8) and $U_f \left(C_t, \tilde{x}_{f,t}, h_t \right)$ is the postdecision state value function associated with DU f.

The update of the postdecision state-value function can be shown using backward induction, as shown in [22]. We refer the interested reader to the proof in [22] for more details. ∎

REFERENCES

[1] M. van der Schaar and S. Shankar, "Cross-layer wireless multimedia transmission: Challenges, principles, and new paradigms," *IEEE Wirel. Commun. Mag.*, vol. 12, no. 4, pp. 50–58, Aug. 2005.

[2] R. Berry and R. G. Gallager, "Communications over fading channels with delay constraints," *IEEE Trans. Inf. Theory*, vol. 48, no. 5, pp. 1135–1149, May 2002.

[3] W. Chen, M. J. Neely, and U. Mitra, "Energy-efficient transmission with individual packet delay constraints," *IEEE Trans. Inform. Theory*, vol. 54, no. 5, pp. 2090–2109, May 2008.

[4] E. Uysal-Biyikoglu, B. Prabhakar, and A. El Gamal, "Energy-efficient packet transmission over a wireless link," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 487–499, Aug. 2002.

[5] M. Goyal, A. Kumar, and V. Sharma, "Optimal cross-layer scheduling of transmissions over a fading multiacess channel," *IEEE Trans. Inform. Theory*, vol. 54, no. 8, pp. 3518–3536, Aug. 2008.

[6] M. van der Schaar and D. Turaga, "Cross-layer packetization and retransmission strategies for delay-sensitive wireless multimedia transmission," *IEEE Trans. Multimedia*, vol. 9, no. 1, pp. 185–197, Jan. 2007.

[7] P. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 390–404, Apr. 2006.

[8] A. Faridi and A. Ephremides, "Distortion control for delay-sensitive sources," *IEEE Trans. Inform. Theory*, vol. 54, no. 8, pp. 3399–3411, Aug. 2008.

[9] A. Banihashemi and A. Hatam, "A distortion optimal rate allocation algorithm for transmission of embedded bitstreams over noisy channels," *IEEE Trans. Commun.*, vol. 56, no. 10, pp. 1581–1584, Oct. 2008.

[10] H. Wang and A. Ortega, "Rate-distortion optimized scheduling for redundant video representations," *IEEE Trans. Image Process.*, vol. 18, no. 2, pp. 225–240, Feb. 2009.

[11] G. Liebl, M. Kalman, and B. Girod, "Deadline-aware scheduling for wireless video streaming," in *Proc. IEEE ICME*, Jul. 2005, p. 4.

[12] C. De Vleeschouwer and P. Frossard, "Dependent packet transmission policies in rate-distortion optimized media scheduling," *IEEE Trans. Multimedia*, vol. 9, no. 6, pp. 1241–1258, Oct. 2007.

[13] F. Fu and M. van der Schaar, "A new systematic framework for autonomous cross-layer optimization," *IEEE Trans. Veh. Tech.*, vol. 58, no. 4, pp. 1887–1903, May 2009.

[14] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Belmont, MA: Athena Scientific, 2005.

[15] J. R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 559–571, Sep. 1994.

[16] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[17] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[18] Q. Zhang and S. A. Kassam, "Finite-state Markov model for Reyleigh fading channels," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1688–1692, Nov. 1999.

[19] A. Reibman and A. Berger, "Traffic descriptors for VBR video teleconferencing over ATM networks," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 329–339, Jun. 1995.

[20] D. Bertsekas and R. Gallager, *Data Networks*. Upper Saddle River, NJ: Prentice-Hall, 1987.

[21] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall.

[22] F. Fu and M. van der Schaar, "Structure-aware stochastic control for transmission scheduling," Univ. California Los Angeles, Los Angeles, Tech. Rep. 4 [Online]. Available: www.arxiv.org/pdf/1003.2471

[23] M. Dai, D. Loguinov, and H. Radha, "Rate distortion modeling for scalable video coders," in *Proc. ICIP*, vol. 2. 2004, pp. 1093–1096.

[24] F. Fu and M. van der Schaar, "Decomposition principles and online learning in cross-layer optimization for delay-sensitive applications," *IEEE Trans. Signal Process.*, vol 58, no. 3, pp. 1401–1415, Feb. 2010.

[25] Y. Eisenberg, C. E. Luna, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Joint source coding and transmission power management for energy-efficient wireless video communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 411–424, Jun. 2002.

[26] A. Dua, C. W. Chan, N. Bambos, and J. Apostolopoulos, "Channel, deadline, and distortion (CD2) aware scheduling of video streams over wireless," *IEEE Trans. Wirel. Commun.*, vol. 9, no. 3, pp. 1001–1011, Mar. 2010.

Fangwen Fu (M'10) received the Bachelors and Masters degrees from Tsinghua University, Beijing, China, in 2002 and 2005, respectively, and the Ph.D. degree from the Department of Electrical Engineering, University of California, Los Angeles, in 2010.

He is currently with Intel, Orangevale, CA, as a Media Architect. He was an Intern with the IBM T. J. Watson Research Center, Yorktown Heights, NY, and with the DOCOMO USA Labs, Palo Alto, CA. His current research interests include wireless multimedia streaming, resource management for networks and systems, stochastic optimization, applied game theory, media hardware architecture, video coding, processing, and analysis.

Dr. Fu received the Dimitris Chorafas Foundation Award in 2009. He was selected by IBM Research as one of the 12 top Ph.D. students to participate in the 2008 Watson Emerging Leaders in Multimedia Workshop in 2008.

Mihaela van der Schaar (F'10) is currently a Professor with the Department of Electrical Engineering, University of California, Los Angeles. She holds 33 granted U.S. patents. Her current research interests include multiuser communication networks, multimedia communications, processing and systems, online learning, network economics, and game theory.

She received the NSF Career Award in 2004, the Best Paper Award from the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2005, the Okawa Foundation Award in 2006, the IBM Faculty Award in 2005, 2007, and 2008, and the Most Cited Paper Award from the *EURASIP: Image Communications Journal* in 2006. She was an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA, IEEE SIGNAL PROCESSING LETTERS, CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and IEEE SIGNAL PROCESSING MAGAZINE. She has received three ISO awards for her contributions to the MPEG video compression and streaming international standardization activities.