# Delay–Rate–Distortion Optimized Rate Control for End-to-End Video Communication Over Wireless Channels

Chenglin Li, *Member, IEEE*, Hongkai Xiong, *Senior Member, IEEE*, and Dapeng Wu, *Fellow, IEEE*

***Abstract*— In addition to the rate-distortion (R-D) behavior, in a real-time wireless video communication system, the end-to-end delay would also significantly affect the overall video reception quality. To analyze, control, and optimize the R-D behavior under the end-to-end delay constraint, in this paper we extend the traditional R-D optimization (RDO) for the wireless video communication system and formulate a novel delay–RDO-based rate control problem, by investigating the allocation of end-to-end delay to different delay components. It aims at minimizing the average total end-to-end distortion under the transmission rate and end-to-end delay constraints, by a joint selection of both the source coding and the channel coding parameters. The wireless channel is represented by a finite-state Markov channel model characterizing the time-varying process and predicting the future channel condition. As applicable solutions, a practical algorithm based on the Lagrange multiplier approaches, Karush–Kuhn–Tucker conditions, and sequential quadratic programming methods is developed. The experimental results demonstrate the superiority of the proposed algorithm over the existing schemes.**

***Index Terms*— Delay–rate–distortion optimization (dRDO), end-to-end distortion, rate control, wireless video.**

## I. Introduction

With the extensive growth of global mobile data traffic and the widespread use of smart devices, wireless video communication applications, such as video surveillance, mobile video services, and consumer electronics multimedia systems, are becoming increasingly popular and have received much attention. According to the Cisco visual networking index [1], mobile video traffic took more than 50% of the total traffic by the end of 2013 and will increase 14-fold between 2013 and 2018, accounting for over two-thirds of the worlds mobile data traffic by the end of 2018. However, transmitting video streams via wireless channels with guaranteed quality of service (QoS) metrics (e.g., end-to-end distortion, and end-to-end delay) is still a challenging problem since the wireless channels are time varying and error prone with respect to wired transmissions.

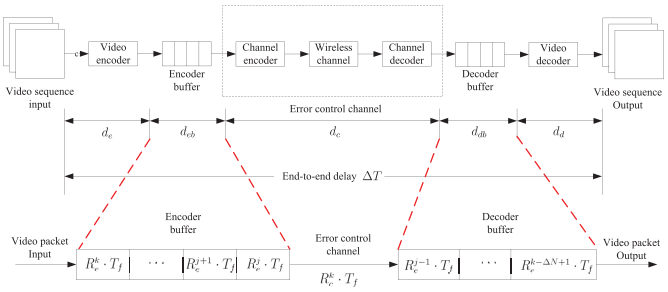


Fig. 1. End-to-end wireless video communication system with end-to-end delay components.

A typical end-to-end wireless video communication system, as shown in Fig. 1, consists of three parts: 1) the video encoder with encoder buffer; 2) the video decoder with decoder buffer; and 3) the error control channel, which can be further decomposed into channel encoder, wireless channel, and channel decoder [2], [3]. Based on the delay–rate–distortion (d–R-D) criterion, such system can be designed to minimize the end-to-end distortion subject to the transmission bit rate constraint and the end-to-end delay bound, by jointly choosing both source coding and channel coding parameters. Though simple, the joint source and channel coding optimization framework for one-hop video transmission provides a theoretical basis and a guideline for the cross-layer design and performance optimization in practical networks with multihop topology. Due to its importance, the point-to-point video streaming scenario has drawn extensive research interest and been widely investigated in [2] and [4]–[7].

As will be discussed in Section II-A, the overall end-to-end distortion of a wireless video communication system includes both the source coding distortion incurred by quantization error and the transmission distortion caused by transmission error. From the perspective of video encoding and regardless of its relationship with the subsequent transmission, to reduce the source coding distortion at the video encoder, we need to enlarge the available source coding bit rate based on Shannon's source coding theory [8]. Since motion estimation (ME) takes the majority of the entire encoding complexity,

Manuscript received April 26, 2014; revised August 2, 2014, October 20, 2014, and December 29, 2014; accepted January 26, 2015. Date of publication February 2, 2015; date of current version September 30, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant U1201255, Grant 61271218, Grant 61228101, and Grant 61425011, and in part by the National Science Foundation under Grant ECCS-1002214 and Grant CNS-1116970. This paper was recommended by Associate Editor E. Steinbach.

C. Li and H. Xiong are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: lcl1985@sjtu.edu.cn; xionghongkai@sjtu.edu.cn).

D. Wu is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: wu@ece.ufl.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2015.2397232

the ME accuracy could be improved with higher encoding complexity [9]. Therefore, a longer encoding time (delay) is also preferred to achieve a lower source coding distortion and thus enhance the overall rate–distortion (R-D) performance of the video encoder. When the subsequent transmission is considered, however, increasing source coding bit rate may result in higher packet error rate for a given channel transmission bit rate and extending encoding time (delay) might lead to more packets dropped due to the end-to-end delay bound violation, which both increase the transmission distortion and would thus affect the overall end-to-end distortion in a negative way.

As implied by these two conflicting aspects, there exist two tradeoffs among delay, rate, and distortion dimensions, which need to be addressed for the end-to-end wireless video communication system design. First, consider the tradeoff between available source coding rate and redundant rate incurred by channel coding, which is controlled by the rate allocation between source and channel coding (or specifically, the channel code rate $r$). If $r$ is increased to allow more available source coding bit rate and thus smaller source coding distortion at the video encoder, the packet error rate and the transmission incurred distortion would also be increased. On the other hand, although the packet error rate and the transmission incurred distortion could be reduced with the reduction of $r$, the available source coding rate is limited and thus the source coding distortion is increased. Considering the total end-to-end distortion comprising both source coding and transmission distortion, an optimum channel code rate $r$ is required to be determined for the rate allocation between source and channel coding.

Next, we will discuss the tradeoff among the end-to-end delay allocation to different delay components. To achieve the optimal end-to-end QoS performance, the entire cross-layer wireless video communication system is expected to appropriately assign different delay components according to the total end-to-end delay bound. Specifically, for a practical real-time wireless video communication system, the end-to-end delay $\Delta T$ experienced by each frame is composed of several delay components, which, as shown in Fig. 1, are video encoding delay $d_e$, encoder buffer delay $d_{\mathrm{eb}}$, channel transmission delay $d_c$ (including channel coding, transmission, and channel decoding delay), decoder buffer delay $d_{\mathrm{db}}$, and video decoding delay $d_d$ [3], [10]. For a given end-to-end delay bound, if the encoding time is increased to achieve better compression performance with higher bit rate, the allowed queueing delay at encoder and decoder buffers will decrease accordingly, which in turn reduces the delay constrained transmission throughput and increases the error incurred transmission distortion. Therefore, the overall system performance depends on the allocation of end-to-end delay to different delay components, and the change of delay assignment in one component would affect the delay budget in other components, thereby impacting the overall system performance.

### A. Related Work

Many rate control schemes have been proposed in the literature to guarantee the QoS metrics for an end-to-end wireless video communication system based on the rate–distortion optimization (RDO). Most of them derive the end-to-end distortion as functions of bit rate and packet error rate [2], [4], [5], while others adopt operational R-D functions [11]. Compared with the operational models that require the video encoder to get all operational functions for different video statistics and channel conditions before the actual encoding, the analytical models are more desirable due to their low complexity [2], [4], [5]. These analytical models cover the analysis of a complete video transmission system, including the R-D performance of the video encoder, forward error correction (FEC), and the effect of error concealment and inter-frame error propagation at the video decoder. Based on the statistical analysis of error propagation, error concealment, and channel decoding, the corresponding RDO-based rate control schemes develop a theoretical framework to estimate the source coding bit rate and the end-to-end distortion and derive an analytical solution for joint source-channel rate control under time-varying wireless channel conditions. However, these schemes are usually designed based on RDO that aims at minimizing the end-to-end distortion while satisfying the transmission rate constraint and neglect the impact of the end-to-end delay constraint on the overall system QoS performance. Since the video packets which arrive too late at the decoder to be decoded before the scheduled display time and thus violate the end-to-end delay bound are useless and considered dropped, serious R-D performance degradation might occur in these schemes if applied to real-time applications and when end-to-end delay bound is taken into consideration.

Some works have been done to extend the traditional R-D model for the video encoder by incorporating the statistical analysis of encoding delay and power [9], [10], [12]. In these works, the analytical framework for delay–power–R-D (d–P–R-D) modeling of the video encoder is proposed, with four dimensions (delay, power, rate, and distortion) derived as functions of source coding parameters, i.e., the search range in ME and the quantization step size. Based on the proposed d–P–R-D model, a source rate control problem could be formulated for the video encoder, which aims at minimizing the source coding distortion under the simple constraints of maximum encoding delay, source coding rate, and encoding power. To achieve the optimal end-to-end QoS performance for the entire end-to-end wireless video communication system, as shown in Fig. 1, however, not only the source coding distortion needs to be minimized but also the transmission distortion should be considered. In addition to the simple constraint of maximum encoding delay, a much more complicated end-to-end delay constraint should also be satisfied, which requires the suitable assignment among the encoding delay and other delay components in accordance with the end-to-end delay bound. In addition, a joint source coding rate and transmission rate constraint needs to be formulated for the determination of channel code rate in accordance with both the video statistics and the channel capacity. For example, as will be shown in Section V-B, if the rate control algorithm in [9] is applied to the video encoder and regardless of transmission error, larger source

coding rate and longer encoding delay are preferred to achieve minimum source coding distortion. By doing so, however, more video packets are likely to be dropped during transmission with a larger transmission distortion incurred due to constrained channel capacity. In this case, the total end-to-end distortion including both source coding and transmission distortion may not be minimum. Therefore, the analytical d–P–R-D model and model-based rate control in [9] are only investigated for the video encoder, but not applicable for the rate control in end-to-end wireless video communication system with end-to-end QoS performance objective and much more complicated delay and rate constraints.

For the end-to-end video communication system, many cross-layer optimized and delay-sensitive video streaming rate control methods have been emerging [3], [6], [7], [13]–[16]. Specifically, in [6], an analytical framework for real-time video communication at the wireless link layer is developed by considering both reliability and latency conditions. In [7], the analytical model for delay-sensitive video transmission is derived and analyzed in a cross-layer optimization framework. Pudlewski *et al.* [14] examine and analyze the wireless video communication at each layer of the networking stack from a cross-layer prospective. To study the end-to-end QoS performance under both channel transmission rate constraint and end-to-end delay bound, Hsu *et al.* [3], [13] formulate the RDO-based rate control problem by translating the end-to-end delay constraints into bit rate constraints at the encoder. In this way, the optimal R-D performance can be achieved while the end-to-end delay bound is satisfied. However, these works usually assume that the video encoding time is constant and thus neglect the tradeoff among the allocation of the end-to-end delay to different delay components. In addition, they suffer two other drawbacks. First, the proposed rate control problem in [3] and [13] is dedicated to only H.261 video codec with no analytical end-to-end distortion model provided, which does not comply with the state-of-the-art video coding standard. In addition, automatic repeat requestis adopted as the transmission error control technique in these works, which is applicable for the two-way communication system where the feedback channel is available. For the wireless mobile channels without additional feedback channels, hence, we need to investigate the QoS performance of the rate control problem with FEC introduced and accordingly seek the optimal tradeoff between the available source coding rate and the redundant rate incurred by channel coding.

### B. Proposed Research

To tackle the above issues of the end-to-end wireless video communication system, in this paper, we extend the traditional RDO and formulate a novel delay–rate–distortion optimization (dRDO)-based rate control problem, with a consideration of the end-to-end delay allocation to different delay components. Based on the analytical source coding d–P–R-D model proposed in [9] and the transmission distortion function derived in [17] and [18], the proposed dRDO based rate control problem aims at minimizing the average total end-to-end distortion (including both the source coding distortion and the transmission distortion) under the transmission rate and end-to-end delay constraints, by a joint selection of the source coding parameters (i.e., the search range in ME and the quantization step size) and the channel coding parameters (i.e., the channel code rate). For the error control channel, the Reed–Solomon (RS) block code is used for FEC. To characterize the time-varying process and predict the future channel condition, the wireless channel is represented by a finite-state Markov channel (FSMC) model. Two kinds of tradeoffs with regard to source coding delay versus buffering delay and available source coding rate versus redundant rate incurred by channel coding are coupled in the proposed dRDO-based rate control problem. To efficiently solve this problem, a practical algorithm based on the Lagrange multiplier approaches, Karush–Kuhn–Tucker (KKT) conditions, and sequential quadratic programming (SQP) methods is also developed.

### C. Paper Organization

The rest of this paper is organized as follows. In Section II, we describe our system model and formulate accordingly the dRDO-based rate control problem. In Section III, we derive the delay, rate, and distortion models for source coding. In addition, the transmission distortion function is derived with RS block code used for FEC and the wireless channel represented by an FSMC model. To achieve minimum end-to-end distortion by an optimal selection of source coding and transmission parameters, a practical algorithm applicable for solving the proposed dRDO based rate control problem is developed in Section IV. Section V presents the experimental results, and evaluates the performance of the proposed algorithm (PA) and the comparison with existing algorithms. The conclusion is given in Section VI.

## II. System Description and Problem Formulation

### A. End-to-End Distortion

In terms of mean squared error (MSE) between the original video frame and the respective reconstructed frame at the decoder, end-to-end distortion is commonly adopted as the performance measure for wireless video communication [2], [5]. For the $k$th frame, the frame level end-to-end distortion is given by

$$D_{\text{ete}}^{k} = D_{e}^{k} + D_{t}^{k} \tag{1}$$

where $D_e^k$ denotes the source coding distortion caused by quantization error during lossy video compression and $D_t^k$ is the transmission distortion caused by transmission error due to bandwidth fluctuation and packet losses. For the derivation of (1), readers are referred to [5].

### B. End-to-End Delay Components and Constraints

For a practical real-time wireless video communication system, the end-to-end delay $\Delta T$ experienced by each frame is composed of several delay components, which, as shown in Fig. 1, respectively, are video encoding delay $d_e$, encoder buffer delay $d_{\text{eb}}$, channel transmission delay $d_c$ (including channel coding, transmission, and channel decoding delay), decoder buffer delay $d_{\text{db}}$, and video decoding

delay $d_d$. It should be noted that for the point-to-point video communication system, the encoded video packets are only needed to be delivered from the video source to the end user via one-hop transmission. Therefore, we assume in this paper that the network layer delay (e.g., delay caused by routing and path selection) is a small constant and thus could be neglected. On the other hand, when the proposed rate control algorithm is applied to a large-scale video streaming network, the effect of such network layer delay can no longer be neglected and thus should be included in the end-to-end delay model. For such applications, we could slightly change the expression of the total end-to-end delay and add into it one more delay component, the network layer delay. Considering the constant video frame rate that is the same at both the encoder and decoder, the end-to-end delay per frame is required to be less than a maximum acceptable delay interval $\Delta T_{\max}$, which is referred to as the end-to-end delay constraint [3]

$$\Delta T = d_e + d_{\text{eb}} + d_c + d_{\text{db}} + d_d \leq \Delta T_{\max}. \tag{2}$$

In other words, every single frame captured by and entering into the video encoder at time $t$ has to be decoded and available for display before time $t + \Delta T_{\max}$. Those video packets which arrive too late at the decoder to be decoded before their scheduled display time and thus violate the maximum end-to-end delay bound are useless and considered lost.

Next, we will further analyze the impact of each individual delay component. In general, the channel transmission delay may be variable. For instance, in the transmission over shared networks, significant delay variations may occur due to queuing in the network routers. Similarly to [3], however, since the video communication system investigated in this paper transmits video packets only via a point-to-point wireless channel connecting the video encoding base station and the end user, the variation of channel transmission delay is relatively small. Thus, it is reasonable to assume $d_c$ to be constant. Furthermore, in [3] and [6], the video encoding time $d_e$ and decoding time $d_d$ are also both assumed to be constant. In fact, the video decoding time (delay) is much shorter than the encoding time (delay) and can be considered as part of the video encoding time since the encoder has to decode the video sequence as well. Therefore, the video decoding delay is negligible compared with the encoding delay, and $d_e + d_d$ can be approximated by $d_e$. However, the video encoding time (delay) is determined by the video encoding complexity, while the video encoding complexity would affect the source coding incurred distortion and bit rate, as verified in the complexity–R-D model of video coding [19]. Consequently, the encoding time controls the distortion and bit rate of the compressed video that is transmitted over the channel. As will be discussed in Section II-D, on the other hand, given an end-to-end delay constraint, if the encoding time is increased to achieve better compression performance with higher bit rate, the allowed queueing delay at encoder and decoder buffers will decrease accordingly. In this case, more packets are likely to be dropped due to delay bound violation, which in turn reduces the delay constrained transmission throughput and thus increases the transmission distortion of the video. In general, for a given end-to-end delay constraint, the overall system performance depends on the allocation of end-to-end delay to different delay components, and the change of delay assignment in one component would affect the delay budget in other components, thereby impacting the overall system performance.

Based on the above assumption and discussion, in this paper, we focus on the assignment and tradeoff between the encoding delay $d_e$ and the buffer delay $d_{\text{buffer}}$, which is defined as the sum of both encoder and decoder buffer delay. Specifically, since $d_c$ is assumed to be constant and $d_d$ is neglected compared with $d_e$, the end-to-end delay constraint is reformulated as

$$d_e + d_{\text{buffer}} \leq \Delta T_{\max} - d_c. \tag{3}$$

Let $T_f$ be the time duration of a frame interval, which can also be considered as the reciprocal of the constant frame rate. Since the time spent by each frame to stay in the encoder buffer and the decoder buffer is $d_{\text{buffer}} = d_{\text{eb}} + d_{\text{db}}$, similar to the estimation adopted for the video transmission in [3], the number of video frames stored either in the encoder buffer or in the decoder buffer is given by

$$\Delta N = \left\lfloor \frac{d_{\text{buffer}}}{T_f} \right\rfloor = \left\lfloor \frac{d_{\text{eb}} + d_{\text{db}}}{T_f} \right\rfloor. \tag{4}$$

In addition, for real-time video communication applications, the video encoder has to encode and output video frames at frame rate $1/T_f$, which makes the video encoding time for each frame $d_e$ constrained by $T_f$. Conversely, if $d_e$ is greater than $T_f$, then the actual output frame rate by the video encoder would be less than the required frame rate $1/T_f$ and thus the accumulated encoding delay $d_e - T_f$ would be introduced for each encoded frame, which causes the real-time encoding impossible. Therefore, to ensure real-time video encoding without introducing accumulated encoding delay for each frame, the maximum video encoding time at the encoder should not exceed the time duration of a frame interval

$$d_e \leq T_f. \tag{5}$$

### C. Source Coding and Transmission Rate Constraints

Due to wireless channel fading, the channel state between the video encoding base station and the end user might change frequently. In accordance with such time-varying characteristics of wireless channels, we assume that time is slotted as $t \in \{1, 2, 3, \ldots\}$ with each slot corresponding to each frame interval and the channels hold their states within the duration of a time slot [20]. Such time-varying channel state information can be captured either through direct measurement (if the duration of a time slot is sufficiently long compared with the required measurement time) or through the combination of measurement and channel state prediction.

To avoid packet drops incurred by delay bound violation, Hsu *et al.* [3] have shown that for a given encoding delay $d_e$, the end-to-end delay constraint can be translated into several constraints on both the encoding rate and the transmission

rate. Assume that at time $t = k$, frame $k$ is the video frame currently being encoded, while the packets of frame $j$ are being transmitted by the error control channel. Let $R_e^i$ be the encoded source bit rate of the $i$th frame, $r$ represent the code rate of channel code, and $R_c^i$ denote the maximum transmission bit rate that is supported by the wireless channel capacity at time $i$. As shown in Fig. 1, at time $t = k$, the encoder buffer contains the packets from frame $j$, $j+1, \ldots, k$, respectively. Therefore, the condition for these packets to arrive at the decoder before the due time is given by

$$\begin{aligned} R_e^j \cdot T_f &\le \sum_{i=k}^{j+\Delta N-1} r \cdot R_c^i \cdot T_f \\ R_e^{j+1} \cdot T_f + R_e^j \cdot T_f &\le \sum_{i=k}^{j+\Delta N} r \cdot R_c^i \cdot T_f \\ &\vdots \\ R_e^k \cdot T_f + \cdots + R_e^{j+1} \cdot T_f + R_e^j \cdot T_f &\le \sum_{i=k}^{k+\Delta N-1} r \cdot R_c^i \cdot T_f. \end{aligned} \tag{6}$$

### *D. Problem Formulation*

From (3) and (4), it can be observed that $\Delta N$ depends on different choices of the encoding delay $d_e$. Intuitively, from the perspective of video coding, a larger encoding delay $d_e$ is preferred to achieve a better R-D performance. By doing so, however, the number of frames stored in the encoder and decoder buffers $\Delta N$ is decreased, which in turn decreases the bit rate that can be transmitted for frames $j$, $j+1, \ldots, k$ according to the rate constraints in (6). Therefore, in the following dRDO-based rate control problem, we aim at achieving the minimum average end-to-end distortion for a video sequence by optimally assigning delay interval for video encoding and buffering as well as allocating the optimum source bit rate for each video frame

$$\min \ \frac{1}{K} \sum_{k=1}^{K} \left[ D_e^k + D_t^k \right] \tag{7a}$$

$$\text{s.t.} \quad d_e + d_{\text{buffer}} \le \Delta T_{\max} - d_c \tag{7b}$$

$$\sum_{i=1}^{k} R_e^i T_f \le \sum_{i=1}^{k+\Delta N-1} r R_c^i T_f \quad \forall k = 1, 2, \ldots, K \tag{7c}$$

$$d_e \le T_f \tag{7d}$$

where $K$ is the number of frames involved for a specific video sequence, and we also assume that $d_e$ and $d_{\text{buffer}}$ can only be adjusted in the sequence level to guarantee that the encoding time and buffering time for each frame is identical such that both the encoder and decoder can have the same constant frame rate. Compared with the existing works on rate control problem in a wireless video communication system [3], [5], the proposed optimization problem in (7) is constrained by one more condition of the encoding delay and the buffer delay in addition to the rate.

It should be noted that (7) is proposed as a general problem formulation for the dRDO-based rate control problem. For the purpose of illustration, here we take the sequence-level rate control problem as an example to give a rough idea of the problem formulation. Using the analytical models provided in Section III, the end-to-end distortion for each frame $k$ in (7a) can be estimated for each possible combination of search range, quantization step size, and channel code rate before encoding the video sequence based on the video statistics and predicted channel condition. Therefore, the analytical minimum average end-to-end distortion can be achieved for the whole video sequence by solving the general optimization problem (7). Here, for the sequence-level rate control problem, $d_e$ and $d_{\text{buffer}}$ can only be adjusted in the sequence level to guarantee the same constant frame rate at both encoder and decoder. As will be discussed and formulated in greater detail in Section IV, the proposed rate control problem (7) can be applied to any desired coding unit, such as a sequence or a group of pictures (GOP), depending on the estimation accuracy of the delay, rate, and distortion models in (7). To be more appropriate for real-time video streaming, we will formulate in Section IV the GOP level rate control problem, where the minimum estimated end-to-end distortion averaged over a GOP with only several frames is achieved by adjusting the source and channel coding parameters for each frame within the GOP.

## III. d–R-D Models for Source Coding and Transmission

The bidirectional B-frames require prediction from both the previous and the subsequent frames, which might incur an additional large delay for encoding and decoding. Due to the stringent end-to-end delay requirement of the wireless video communication systems, the IBPBP coding mode with bidirectional prediction B-frames will not be allowed for delay-constrained video applications, such that both the encoding and the decoding order is strictly causal [16]. For this reason, in most existing works on real-time wireless video communication that requires low end-to-end delay, the video encoding mode is adopted as IPPPP coding mode [6], [16]. Therefore, in this section, we derive the delay, rate, and distortion functions for source coding and the transmission distortion function for IPPPP coding mode.

### *A. d–R-D Source Coding Model*

In [5], and [9], we have theoretically derived the formulas of source coding delay, rate, and distortion for the IPPPP coding mode in H.264/AVC. Under the assumption that the transformed residuals in ME follow the Laplacian distribution [21], [22], both the source rate and distortion of an inter-coded P-frame $k$ are derived as functions of the standard deviation $\sigma^k$ of the transformed residuals and the quantization step size $Q^k$. Specifically, the source rate function is approximated by the entropy of the quantized transformed residuals, and the source distortion is only incurred by the

quantization error as

$$R_e(\Lambda^k, Q^k) = -P_0 \log_2 P_0 + (1-P_0) \times \left[\frac{\Lambda^k Q^k \log_2 e}{1-e^{-\Lambda^k Q^k}} - \log_2(1-e^{-\Lambda^k Q^k}) - \Lambda^k Q^k \gamma \log_2 e + 1\right] \quad (8)$$

$$D_e(\Lambda^k, Q^k) = \frac{\Lambda^k Q^k e^{\gamma \Lambda^k Q^k}(2+\Lambda^k Q^k - 2\gamma \Lambda^k Q^k) + 2 - 2e^{\Lambda^k Q^k}}{(\Lambda^k)^2(1-e^{\Lambda^k Q^k})} \quad (9)$$

where $\Lambda^k = \sqrt{2}/\sigma^k$ is the Laplace parameter that is one-to-one mapping of $\sigma^k$, $\gamma Q^k$ represents the rounding offset and $\gamma$ is a parameter between $(0, 1)$, such as $1/6$ for H.264/AVC inter-frame coding [21], and $P_0 = 1 - e^{-\Lambda^k Q^k(1-\gamma)}$ is the probability of quantized transform coefficient being zero. For a specific video sequence, $\sigma^k$ can be well fitted by a closed-form function of the search range $\lambda^k$ in ME and the quantization step size $Q^k$ [9] as

$$\sigma^k(\lambda^k, Q^k) = ae^{-b\lambda^k} + c + dQ^k \quad (10)$$

where $a$, $b$, $c$, and $d$ are fitting parameters dependent on the encoded video sequence as well as on the encoding structure. Therefore, integrating $\Lambda^k = \sqrt{2}/\sigma^k$ into (8) and (9), both source coding rate and distortion of the $k$th frame can be further expressed as functions of $\lambda^k$ and $Q^k$, i.e., $R_e^k(\lambda^k, Q^k)$, and $D_e^k(\lambda^k, Q^k)$, respectively.

It has also been justified by [9] that the encoding delay for an inter-coded P-frame can be approximated by the ME time, since ME takes the majority of the entire encoding time. Specifically, the ME time is derived as the total number of CPU clock cycles consumed by its sum of absolute difference (SAD) operations divided by the number of clock cycles per second. Thus, for the single-reference prediction case where only one reference frame is used for ME of the current frame, the encoding delay of the $k$th frame can also be expressed as a function of $\lambda^k$ and $Q^k$ as

$$d^k(\lambda^k, Q^k) = \frac{N(2\lambda^k+1)^2 \cdot \upsilon(Q^k) \cdot c_0}{f_{\mathrm{CLK}}} \quad (11)$$

where $N$ is the number of MBs in a frame, $(2\lambda^k+1)^2 \cdot \upsilon(Q^k)$ is the total number of SAD operations in the 2-D search area for each macroblock (MB) and $\upsilon(Q^k)$ denotes the ratio of the actual number of SAD operations in the joint model (JM) codec to the theoretical total number of SAD operations, $c_0$ is the number of clock cycles of one SAD operation over a given CPU, and $f_{\mathrm{CLK}}$ is the constant clock frequency of the CPU.

### B. Derivation of Transmission Distortion

As derived in [17] and [18], for single-reference ME and no slice data partitioning, the frame-level transmission distortion for the $k$th frame is given by

$$D_t^k(\bar{P}^k) = \bar{P}^k\left(E[(\varepsilon^k)^2] + \rho^k E[(\xi^k)^2] + D_t^{k-1}\right) + (1-\bar{P}^k)\alpha^k D_t^{k-1} \quad (12)$$

where $\bar{P}^k$ is the average packet error probability (PEP) of all packets within the $k$th frame; $\varepsilon^k$ and $\xi^k$ denote the residual concealment error and motion vector concealment error, respectively; the propagation factor $\alpha^k$ and the correlation ratio $\rho^k$ are the system parameters that depend on the video content, channel condition, and codec structure; and $D_t^{k-1}$ is the transmission distortion of frame $k-1$. From the analysis in [18], $E[(\varepsilon^k)^2]$ can be estimated by $[\sigma^k(\lambda^k, Q^k)]^2$. As will be introduced in detail in Appendix A, $E[(\xi^k)^2]$ can also be fitted as a closed-form function, $MV_e(\lambda^k, Q^k)$. In addition, the derivation process of the propagation factor $\alpha^k$ and the correlation ratio $\rho^k$ is given in [18]. Therefore, it can be seen that the transmission distortion is basically a function of the search range, quantization step size, and PEP, while the other video frame statistics and system parameters could be simply estimated, as discussed in [18]. Therefore, in the following, we will describe how to estimate $\bar{P}^k$ for each frame.

*1) Error Control Channel:* By introducing FEC into video communication systems, the reliability of transmission is improved with smaller PEP and thus lower transmission distortion. To enhance the error correction capability, however, redundant protection information is also introduced. As shown in the rate constraints (6), to maintain the transmission data rate $R_c$, with channel coding, the maximum achievable source data rate for video encoder has to be reduced to $R_e \le rR_c$, where $r$ is the channel code rate between the range of $[0, 1]$. For a given channel data rate $R_c$, the code rate $r$ controls the bit rate allocation between source and channel coding, and thus the tradeoff between source coding and transmission incurred distortion. Specifically, a smaller $r$ is preferred for the purpose of reducing the PEP as well as the transmission distortion. By lowering $r$, however, the available source coding rate is also decreased, which will result in higher source coding distortion. Since the optimization objective is minimizing the total distortion of both source coding and transmission distortion, it is required to find the best tradeoff between these two types of distortion by choosing a proper code rate $r$.

In this paper, we assume that the $(n, m)$ RS block code is used for FEC with a block of $m$ information symbols and $n-m$ parity symbols. Here, we use the common choice of 8 b per symbol, and thus one symbol corresponds to one byte. The code rate is therefore $r = m/n$, and the maximum block length is $n_{\max} = 2^8 - 1 = 255$. In the proposed video communication system, we set the number of symbols within each encoded video packet less than $n_{\max}$, and thus each video packet can be considered as the information symbols and be encoded into one coded block by RS coding. In this way, a video packet after source coding corresponds to an RS code block in channel transmission. With the $(n, m)$ RS code, any received block with symbol errors less than $t_c = \lfloor (n-m)/2 \rfloor$ can be successfully recovered. Thus, the probability of a block

(video packet) being unable to correct is given by

$$P = \sum_{\kappa=t_c+1}^{n} P_b(n,\kappa) \tag{13}$$

where $P_b(n,\kappa)$ is the probability that $\kappa$ symbol errors occur within a block of $n$ consecutively transmitted symbols. Further denoting $P_s$ by the symbol error probability of the wireless channel, $P_b(n,\kappa)$ is therefore given by

$$P_b(n,\kappa) = \binom{n}{\kappa} P_s^{\kappa}(1-P_s)^{n-\kappa}. \tag{14}$$

*2) Finite-State Markov Channel Model:* According to [23]–[25], a good approximation in modeling the time-varying error process of a wireless channel can be provided by the ergodic FSMC model. Suppose that the wireless channel has a finite set of $H$ states corresponding to $H$ different symbol error probabilities, denoted by the vector $\mathbf{P}_s = (P_s^1, \ldots, P_s^h, \ldots, P_s^H)^T$, and a transition probability matrix $\mathbf{T} \in \mathbb{R}_+^{H\times H}$ that has the following structure:

$$\mathbf{T} = \begin{pmatrix} t_{1,1} & t_{1,2} & 0 & 0 & 0 & \ldots & 0 & 0 \\ t_{2,1} & t_{2,2} & t_{2,3} & 0 & 0 & \ldots & 0 & 0 \\ 0 & t_{3,2} & t_{3,3} & t_{3,4} & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \ldots & t_{H,H-1} & t_{H,H} \end{pmatrix} \tag{15}$$

where $t_{i,j}$ is the transition probability from state index $i$ to $j$ and can be determined for the Rayleigh fading channel, as in [24]. Such channel models (Rayleigh fading channel or slow fading channel) are also adopted by the recent works on the video streaming applications [26], [27]. Suppose that at time $t = k$, we have observed the channel state with an associated symbol error probability as $P_s(t=k) = P_s^h$. Based on the transition probability that is independent of time slot, the expected symbol error probability of that channel at time slot $t = k+1$, given the information is available at time slot $t = k$, can be predicted as

$$P_s(t=k+1|t=k) = \mathbf{T}_{(h)} \cdot \mathbf{P}_s = t_{h,h} P_s^h + t_{h,h-1} P_s^{(h-1)} + t_{h,h+1} P_s^{(h+1)} \tag{16}$$

where $\mathbf{T}_{(h)}$ denotes the $h$th row of $\mathbf{T}$. By recursively using (16), all the expected values of future symbol error probabilities can be predicted by the $h$th row of $\Delta k$th power of $\mathbf{T}$

$$P_s(t=k+\Delta k|t=k) = \mathbf{T}_{(h)}^{(\Delta k)} \cdot \mathbf{P}_s, \quad \Delta k = 1, 2, \ldots \tag{17}$$

Note that in the rate constraints (6), in order for the $k$th frame to arrive at the decoder before the maximum delay bound, all packets belonging to that frame have to be transmitted through the wireless channel between the time interval $[k, k+\Delta N-1]$. Therefore, we can use the average symbol error probability over the time interval $[k, k+\Delta N-1]$ to approximate the average symbol error probability during the transmission of packets for the $k$th frame as

$$\bar{P}_s^k = \frac{1}{\Delta N} \sum_{\Delta k=k}^{k+\Delta N-1} P_s(t=k+\Delta k|t=k). \tag{18}$$

Integrating (18) and (14) into (13), the average PEP of all packets within the $k$th frame can be obtained, which only depends on the future channel state information and the channel code rate $r$. Since the future channel state information can be predicted for the wireless channel based on the current observation, the transmission distortion can be further expressed as a function of the channel code rate as well as the search range and quantization step size, i.e., $D_t^k(\lambda^k, Q^k, r)$.

## IV. dRDO-Based Cross-Layer Rate Control and Algorithm Design

Adapting to different applications, the dRDO-based rate control problem proposed in (7) can be applied to any desired coding unit, e.g., a sequence, or a GOP. For example, if it is applied to an entire video sequence, (7) can be regarded to solve the sequence-level rate control problem. If it is applied to a GOP, (7) can behave as a GOP-level rate control problem. To be more appropriate for real-time video streaming and without loss of generality, in this paper, a GOP-level rate control problem will be imposed on (7) with a practical solution. Specifically, suppose that a GOP with the IPPPP coding structure has one I-frame and $K$ P-frames. The first I-frame within each GOP is indexed as frame 0 and assumed to be encoded with fixed coding parameters. Therefore, the dRDO-based rate control problem proposed in (7) can be reformulated as

$$\min_{\boldsymbol{\lambda},\mathbf{Q},\Delta N,r} \quad \frac{1}{K}\sum_{k=1}^{K}\left[D_e^k(\lambda^k, Q^k) + D_t^k(\lambda^k, Q^k, r)\right] \tag{19a}$$

$$\text{s.t.} \quad d_e^k(\lambda^k, Q^k) + \Delta N T_f \le \Delta T_{\max} - d_c \quad \forall k = 1, 2, \ldots, K \tag{19b}$$

$$\sum_{i=1}^{k} R_e^i(\lambda^i, Q^i) \le \sum_{i=1}^{k+\Delta N-1} r R_c^i \quad \forall k = 1, 2, \ldots, K \tag{19c}$$

$$d_e^k(\lambda^k, Q^k) \le T_f \quad \forall k = 1, 2, \ldots, K \tag{19d}$$

where $\boldsymbol{\lambda} = (\lambda^1, \ldots, \lambda^k, \ldots, \lambda^K)$ and $\mathbf{Q} = (Q^1, \ldots, Q^k, \ldots, Q^K)$ are the vector representations of search ranges and quantization step sizes for all P-frames within the GOP. From the above analysis, two kinds of tradeoffs, regarding source coding delay versus buffering delay and available source coding rate versus channel code rate, are coupled in the optimization problem (19) that targets at minimizing the average total end-to-end distortion. Theoretically, a larger search range $\lambda^k$ as well as a smaller quantization step size $Q^k$ is required to result in smaller source coding distortion with larger source coding bit rate. However, the source coding delay is also enlarged, while the number of frames stored in the encoder and decoder buffers $\Delta N$ is decreased, which in turn limits the available source coding bit rate supported by the transmission channel and thus affects the source coding distortion. Furthermore,

from the perspective of channel coding and transmission, by lowering $r$, the PEP as well as the transmission distortion can be reduced, while a larger $r$ is preferred for source coding to promise a sufficient source coding rate and thus smaller source coding distortion. As a matter of fact, the target of the optimization problem in (19) is to minimize the average end-to-end distortion for the given end-to-end delay constraints and channel transmission rate constraints, by the appropriate selections of source coding parameters $(\boldsymbol{\lambda}, \mathbf{Q})$ as well as system and channel coding parameters $(\Delta N, r)$.

### A. Optimization Decomposition and Approximation

In general, it is difficult to solve optimization problem (19), since the function form of $D_t^k(\lambda^k, Q^k, r)$ is complicated and accumulated over frames, and also because the auxiliary optimization variable $\Delta N$ is included in the upper limit of the summation in (19c). In a practical system design for the end-to-end video communication, the channel code rate $r$ is usually selected from a discrete set $\mathcal{R}$ containing limited number of choices for $r$ values. We also assume, as in [24], that the dynamic wireless channel change can be mainly characterized by the time-varying symbol error probability, and the maximum bit rate supported by the wireless channel capacity over different time slots is identical and denoted by $R_c^i = R_c$. Therefore, by fixing $r = r'$ where $r' \in \mathcal{R}$, optimization problem (19) can be decomposed and approximated as subproblem

$$\min_{\boldsymbol{\lambda}, \mathbf{Q}, \Delta N, r=r'} \quad \frac{1}{K}\sum_{k=1}^{K}\left[D_e^k(\lambda^k, Q^k) + D_t^k(\lambda^k, Q^k, r)\right] \tag{20a}$$

$$\text{s.t.} \quad d_e^k(\lambda^k, Q^k) + \Delta N T_f + d_c - \Delta T_{\max} \le 0 \quad \forall k = 1, \ldots, K \tag{20b}$$

$$\sum_{i=1}^{k} R_e^i(\lambda^i, Q^i) - (k + \Delta N - 1) r R_c \le 0 \quad \forall k = 1, 2, \ldots, K \tag{20c}$$

$$d_e^k(\lambda^k, Q^k) - T_f \le 0 \quad \forall k = 1, 2, \ldots, K. \tag{20d}$$

Note that the reason to implement the decomposition is twofold. First, $r$ is included in the lower limit of the summation in the PEP computation with floor function operation involved, as indicated in (13), while the transmission distortion is a function of the PEP, as shown in (12). Therefore, integrating (13) into (12), the end-to-end transmission distortion would be a very complicated function of $r$, since $r$ not only appears in the lower limit of the summation but also is involved in the floor function. If the value of $r$ is not fixed, it is theoretically unable to solve the optimization problem (19) due to the complicated function form of $D_t^k(\lambda^k, Q^k, r)$ in the objective function (19a). Second, by fixing $r = r' \in \mathcal{R}$, the original optimization problem (19) is transformed into a subproblem (20) that is practically solvable. In addition, the number of subproblems (20) is limited by the number of elements in $\mathcal{R}$, which is usually not large in practical system design. Therefore, we can construct multiple subproblems (20) in accordance with different $r$ values and choose the subproblem achieving the minimum average end-to-end distortion as the global optimal solution of the original optimization problem (19).

Let $D(\boldsymbol{\lambda}, \mathbf{Q})$, $d^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N)$, and $R^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N)$ denote the optimization objective (20a), the inequality constraints in (20b), and (20c), respectively, the subproblem (20) is then summarized as

$$\min_{\boldsymbol{\lambda}, \mathbf{Q}, \Delta N, r=r'} \quad D(\boldsymbol{\lambda}, \mathbf{Q}) \tag{21a}$$

$$\text{s.t.} \quad d^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N) \le 0 \quad \forall k = 1, 2, \ldots, K \tag{21b}$$

$$R^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N) \le 0 \quad \forall k = 1, 2, \ldots, K \tag{21c}$$

$$d_e^k(\lambda^k, Q^k) \le T_f \quad \forall k = 1, 2, \ldots, K. \tag{21d}$$

For the detailed derivation process of $D(\boldsymbol{\lambda}, \mathbf{Q})$, readers are referred to Appendix B. Therefore, for any given channel code rate $r = r', r' \in \mathcal{R}$, we can construct a subproblem (21). If its corresponding optimal solution $(\boldsymbol{\lambda}', \mathbf{Q}', \Delta N')$ can be found, then it is only required to loop for all possible values of $r$ with corresponding optimal solutions and select the one achieving the minimum average end-to-end distortion as the global optimal solution of optimization problem (19). In the next section, we will accordingly develop a practical approach to solving subproblem (21).

### B. Algorithm Design

Either the Lagrange multiplier method [28]–[30] or the dynamic programming approach [31] can be applied to solve subproblem (21). The former is preferred throughout this paper since it can be implemented independently in each coding unit. In comparison, the dynamic programming approach requires a tree representing all possible solutions to grow over multiple coding units. The computational complexity would grow exponentially with the number of coding units, which is not affordable for practical applications. With the Lagrange multiplier method, the subproblem (21) can be converted to an unconstrained problem

$$\begin{aligned}\min \; L(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N, \boldsymbol{\mu}, \boldsymbol{\eta}, \boldsymbol{\varphi}) &= D(\boldsymbol{\lambda}, \mathbf{Q}) + \sum_{k=1}^{K} \mu^k \cdot d^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N) \\ &+ \sum_{k=1}^{K} \eta^k \cdot R^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N) + \sum_{k=1}^{K} \varphi^k \cdot \left[d_e^k(\lambda^k, Q^k) - T_f\right]\end{aligned} \tag{22}$$

where $\boldsymbol{\mu} \ge 0$, $\boldsymbol{\eta} \ge 0$, and $\boldsymbol{\varphi} \ge 0$ are the Lagrange multipliers associated with the inequality constraints (21b), (21c), and (21d), respectively.

In the following, we will propose a practical algorithm for the subproblem (21) based on KKT conditions and SQP methods, which can produce both primal ($\boldsymbol{\lambda}^*$, $\boldsymbol{Q}^*$, and $\Delta N^*$) and dual (Lagrange multipliers $\boldsymbol{\mu}^*$, $\boldsymbol{\eta}^*$, and $\boldsymbol{\varphi}^*$) solutions simultaneously in an iterative way. To solve the first-order necessary conditions of optimality for the subproblem (21), the SQP methods [32], [33] can be utilized to construct a quadratic programming subproblem at a given approximate solution and then to employ the solution to this subproblem to construct a better approximation. This process is iterated to create a sequence of approximations that is expected to converge to the optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{Q}^*, \Delta N^*, \boldsymbol{\mu}^*, \boldsymbol{\eta}^*, \boldsymbol{\varphi}^*)$. Specifically, given an iterate $(\boldsymbol{\lambda}_\tau, \boldsymbol{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_\tau, \boldsymbol{\eta}_\tau, \boldsymbol{\varphi}_\tau)$, a new iterate $(\boldsymbol{\lambda}_{\tau+1}, \boldsymbol{Q}_{\tau+1}, \Delta N_{\tau+1}, \boldsymbol{\mu}_{\tau+1}, \boldsymbol{\eta}_{\tau+1}, \boldsymbol{\varphi}_{\tau+1})$ can be obtained

by solving a quadratic programming (QP) minimization subproblem given by

$$\min_{\delta_\tau} \ \nabla D(\boldsymbol{\lambda}_\tau, \boldsymbol{Q}_\tau)^T \cdot \delta_\tau + \frac{1}{2}\delta_\tau{}^T \cdot \nabla^2 L(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_\tau, \boldsymbol{\eta}_\tau, \boldsymbol{\varphi}_\tau) \cdot \delta_\tau \tag{23a}$$

$$\text{s.t. } \nabla d^k(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T \cdot \delta_\tau + d^k(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau) = 0 \quad \forall k \tag{23b}$$

$$\nabla R^k(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T \cdot \delta_\tau + R^k(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau) = 0 \quad \forall k \tag{23c}$$

$$\nabla d_e^k\left(\lambda_\tau^k, Q_\tau^k\right)^T \cdot \delta_\tau + d_e^k\left(\lambda_\tau^k, Q_\tau^k\right) = T_f \quad \forall k \tag{23d}$$

where the derivative operators $\nabla$ and $\nabla^2$ are used to refer to the first-order gradient vector and the second-order Hessian matrix with respect to primal variables $(\boldsymbol{\lambda}, \boldsymbol{Q}, \Delta N)$, respectively, and $\delta_\tau = (\boldsymbol{\lambda}_{\tau+1} - \boldsymbol{\lambda}_\tau, \boldsymbol{Q}_{\tau+1} - \boldsymbol{Q}_\tau, \Delta N_{\tau+1} - \Delta N_\tau)^T$ is the vector representing the update directions of primal variables.

The aforementioned SQP algorithm, though can be used to appropriately solve (21), suffers two deficiencies similar to the Newton method. First, it requires at each iteration the calculation of second-order Hessian matrix $\nabla^2 L(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_\tau, \boldsymbol{\eta}_\tau, \boldsymbol{\varphi}_\tau)$, which could be a costly computational burden and in addition might not be positive definite. To address this issue, we can use the quasi-Newton method instead to construct an approximate Hessian matrix $B_\tau$ by which $\nabla^2 L(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_\tau, \boldsymbol{\eta}_\tau, \boldsymbol{\varphi}_\tau)$ is replaced in (23). In practice, such an approximation $B_\tau$ can be obtained by the Broyden–Fletcher–Goldfarb–Shanno method [34]

$$B_{\tau+1} = B_\tau + \frac{\gamma_\tau \gamma_\tau{}^T}{\gamma_\tau{}^T \delta_\tau} - \frac{B_\tau \delta_\tau \delta_\tau{}^T B_\tau{}^T}{\delta_\tau{}^T B_\tau \delta_\tau} \tag{24}$$

with $\gamma_\tau$ defined by

$$\gamma_\tau = \nabla L(\boldsymbol{\lambda}_{\tau+1}, \mathbf{Q}_{\tau+1}, \Delta N_{\tau+1}, \boldsymbol{\mu}_{\tau+1}, \boldsymbol{\eta}_{\tau+1}, \boldsymbol{\varphi}_{\tau+1}) - \nabla L(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_{\tau+1}, \boldsymbol{\eta}_{\tau+1}, \boldsymbol{\varphi}_{\tau+1}). \tag{25}$$

Second, to achieve global convergence performance, a line search method [33] is used to replace the full Newton step $(\boldsymbol{\lambda}_{\tau+1}, \mathbf{Q}_{\tau+1}, \Delta N_{\tau+1})^T = (\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T + \delta_\tau$ by $(\boldsymbol{\lambda}_{\tau+1}, \mathbf{Q}_{\tau+1}, \Delta N_{\tau+1})^T = (\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T + \beta \cdot \delta_\tau$, which defines the $l_1$ merit function as

$$l_1(\boldsymbol{\lambda}, \boldsymbol{Q}, \Delta N) = D(\boldsymbol{\lambda}, \boldsymbol{Q}) + \theta \cdot \left[ \sum_{k=1}^{K} \max\{0, d^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N)\} + \sum_{k=1}^{K} \max\{0, R^k(\boldsymbol{\lambda}, \mathbf{Q}, \Delta N)\} + \sum_{k=1}^{K} \max\left\{0, d_e^k(\lambda^k, Q^k)\right\} \right] \tag{26}$$

where $\theta$ is a positive penalty parameter and the step size $\beta$ is chosen such that the $l_1$ merit function is reduced

$$l_1\left((\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T + \beta \cdot \delta_\tau\right) < l_1\left((\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T + \delta_\tau\right). \tag{27}$$

The improved SQP algorithm with the consideration of both of the above modifications can be applied to solve the subproblem (21). If we further loop for all possible values of $r$ with corresponding optimal solutions to the subproblem (21), then the one achieving the minimum average end-to-end distortion can be selected as the global optimal solution of optimization problem (19). Accordingly, the practical algorithm for the dRDO-based rate control problem is given in Algorithm 1.

**Algorithm 1** dRDO Rate Control Algorithm for Optimization Problem (19)

**Initialization Step** (for a given channel code rate $r = r' \in \mathcal{R}$)

Set an initial primal/dual point $(\boldsymbol{\lambda}_0, \mathbf{Q}_0, \Delta N_0, \boldsymbol{\mu}_0, \boldsymbol{\eta}_0, \boldsymbol{\varphi}_0)$ with $\boldsymbol{\mu} \succeq 0$, $\boldsymbol{\eta} \succeq 0$, and $\boldsymbol{\varphi} \succeq 0$, and a positive definite matrix $B_0$.

Let $\tau = 0$, and go to the iteration step.

**Iteration Step**

At $\tau$-th iteration:

1. Solve the quadratic sub-problem (23), with $\nabla^2 L(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_\tau, \boldsymbol{\eta}_\tau, \boldsymbol{\varphi}_\tau)$ replaced by $B_\tau$, to obtain $\delta_\tau$ together with a set of Lagrange multipliers $(\boldsymbol{\mu}_{\tau+1}, \boldsymbol{\eta}_{\tau+1}, \boldsymbol{\varphi}_{\tau+1})$.
2. If $\delta_\tau = 0$, which indicates $(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau, \boldsymbol{\mu}_{\tau+1}, \boldsymbol{\eta}_{\tau+1}, \boldsymbol{\varphi}_{\tau+1})$ satisfies the KKT conditions of sub-problem (21), or $\tau + 1$ exceeds the predefined maximum number of iterations, then go to the decision step.
3. Find $(\boldsymbol{\lambda}_{\tau+1}, \mathbf{Q}_{\tau+1}, \Delta N_{\tau+1})^T = (\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)^T + \beta \cdot \delta_\tau$ according to Eqs. (26) and (27).
4. Update $B_\tau$ to a positive definite matrix $B_{\tau+1}$ based on Eq. (24).
5. Set $\tau = \tau + 1$, and return to step 1.

**Decision Step**

Determine whether all the possible values $r' \in \mathcal{R}$ have been traversed. If no, set $r$ to a new value of which the corresponding sub-problem (21) is not solved, and go to the initialization step. If yes, choose a value $r = r^* \in \mathcal{R}$ with the minimum average end-to-end distortion as the optimal channel code rate, and set the associated $(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau, \Delta N_\tau)|_{r=r^*}$ as the optimal source coding parameters for optimization problem (19).

## V. Experimental Results

In this section, we evaluate the proposed dRDO-based rate control algorithm through both the analysis of the PA itself under different end-to-end system settings and the comparison of the d–R-D performance achieved by the PA to those using existing schemes. Specifically, we implement the PA in JM18.2 [35] codec, with test video sequences *Bus* [quarter common intermediate format (QCIF)], *Foreman* [common intermediate format (CIF)], and *Mobile* (CIF), the IPPPP GOP structure with a GOP containing one I-frame and 32 P-frames, CABAC entropy coding, the maximum search range 16 of the ME, the dynamic ranges 0 to 51 of the quantization parameter, and one reference frame. These three test video sequences correspond to different content types, i.e., camera movement and large object motion for the *Bus* sequence, medium but

TABLE I
FSMC MODEL PARAMETERS IN [24]

| $h$ | $\Gamma_h/\overline{SNR}$ (dB) | $P_s^h$ ($\overline{SNR}=2$dB) | $P_s^h$ ($\overline{SNR}=5$dB) | $P_s^h$ ($\overline{SNR}=10$dB) |
|---|---|---|---|---|
| 1 | $-\infty$ | $8.286\cdot10^{-1}$ | $7.529\cdot10^{-1}$ | $5.570\cdot10^{-1}$ |
| 2 | -12.0474 | $5.797\cdot10^{-1}$ | $4.153\cdot10^{-1}$ | $1.313\cdot10^{-1}$ |
| 3 | -6.0158 | $3.328\cdot10^{-1}$ | $1.555\cdot10^{-1}$ | $9.900\cdot10^{-3}$ |
| 4 | -2.4754 | $1.578\cdot10^{-1}$ | $4.008\cdot10^{-2}$ | $2.429\cdot10^{-4}$ |
| 5 | 0.0499 | $6.157\cdot10^{-2}$ | $7.079\cdot10^{-3}$ | $1.813\cdot10^{-6}$ |
| 6 | 2.0232 | $1.964\cdot10^{-2}$ | $8.840\cdot10^{-4}$ | $3.782\cdot10^{-9}$ |
| 7 | 3.6514 | $5.069\cdot10^{-3}$ | $6.614\cdot10^{-5}$ | $2.003\cdot10^{-12}$ |
| 8 | 5.0454 | $1.042\cdot10^{-3}$ | $3.285\cdot10^{-6}$ | - |
| 9 | 6.2726 | $1.669\cdot10^{-4}$ | $9.867\cdot10^{-8}$ | - |
| 10 | 7.3777 | $2.016\cdot10^{-5}$ | $1.681\cdot10^{-9}$ | - |
| 11 | 8.3934 | $1.449\cdot10^{-6}$ | $1.214\cdot10^{-11}$ | - |
| - | $\infty$ | - | - | - |

complex motion for the *Foreman* sequence, and zooming effects for *Mobile* sequence, respectively. As shown in [9], both the encoding complexity and the encoding time of a video frame would increase quadratically with the resolution of that frame. Therefore, to have comparable single frame encoding time and thus to ensure real-time encoding, we assume in this paper that for the QCIF video sequences, the best inter-mode for a MB in ME can be selected from all the eight possible inter-modes based on RDO, while for CIF video sequences, the MB coding mode is fixed at $16\times16$ inter-mode to shorten the encoding time of a single frame with CIF resolution.

After source coding, each encoded video sequence is further coded by the RS block code with a code rate selected from the discrete set $\mathcal{R}=\{1/8,\ 1/4,\ 3/8,\ 1/2,\ 5/8,\ 3/4,\ 7/8\}$ and transmitted through and tested under different Rayleigh fading channels, i.e., multiple combinations of transmission bit rate $R_c$ from 100 to 500 kb/s and average signal-to-noise ratio (SNR) from 2 to 10 dB. To characterize and represent the time-varying behavior of the Rayleigh fading channels, the FSMC model in [24] is adopted, with SNR thresholds and symbol error probabilities for different channel states listed in [24, Table I]. For completeness of this paper, we include this table here with slight adaptation to our scenario. In Table I, $\overline{\text{SNR}}$ represents the average SNR, $\Gamma_h$ denotes the received SNR threshold, and the channel is in state $h$ if the received SNR locates in the range $[\Gamma_h, \Gamma_{h+1})$.

### A. Proposed Algorithm Analysis

From Algorithm 1, the accuracy of end-to-end distortion computation is critical to the overall performance of the PA. Accordingly, the accuracy of the end-to-end distortion model is evaluated in comparison with the actual MSE measure in Fig. 2. It is observed that the model estimated end-to-end distortion is relatively close to the true end-to-end distortion and thus the model accuracy is verified from the experimental evaluations. For more detail, the derivation process as well as the analysis of model accuracy of the rate, distortion, and encoding time models in (8)–(11) are provided and justified in detail in [9]. As shown in [9], while the source rate model in (8), the source distortion model in (9), and the encoding time model in (11) are general models independent of the video content, only the model parameters of the standard deviation model in (10) are specific to different video scenes. Therefore, a number of frames representing the same video

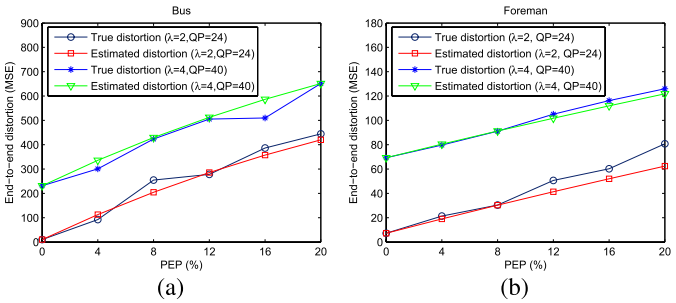


Fig. 2. End-to-end distortion versus PEP for (a) *Bus* sequence and (b) *Foreman* sequence.

scene will have the same set of model parameters in (10). Specifically, when a new video scene is acquired, a new model $\sigma^k(\lambda^k, Q^k) = ae^{-b\lambda^k} + c + dQ^k$ needs to be fitted for all the frames within such video scene with the same set of four fitting parameters $a$, $b$, $c$, and $d$. To have a better fitting result, the whole set of the empirical values with different configurations of $\lambda^k$ and $Q^k$ can be used to determine the four fitting parameters, which may cause the increase in computational complexity. To reduce such complexity, in practice, since the function form of $\sigma^k(\lambda^k, Q^k)$ is already known and only four fitting parameters are unknown, we could choose a much smaller subset of empirical values with only a few configurations of $\lambda^k$ and $Q^k$ as the training set and obtain the standard deviation model. The detailed training set selection process is given in Appendix C. On the other hand, in the PA, we only need to use the first several inter-frames (e.g., 1–10 frames) to obtain the standard deviation model in (10) and to apply it as an estimated standard deviation model for the complete video scene (e.g., 200–300 frames). For example, if we choose a $3\times5$ training subset, as discussed in Appendix C, and apply each of the 15 pairs of $\lambda^k$ and $Q^k$ to only one inter-frame to obtain the fitted model in (10), then the computational time for the model training can be approximated by the time used to encode 15 inter-frames. Therefore, for a video scene with hundreds of frames, the additional computational complexity introduced by the PA for model training per frame is not significant and thus can be neglected.

In practice, two methods can be used to determine whether to update the model parameters in (10). A solution is to adopt the scene-change-detection method [36] in the compressed domain where the discrete cosine transform coefficients are utilized to detect video scene change. As an alternative, during the encoding process, we can compare the actual value of $\sigma^k$ after encoding each frame $k$ with the estimated value by (10). Since the model in (10) is fitted for a specific video scene, if frame $k$ belongs to the same video scene, the difference between the actual value and estimated value of $\sigma^k$ would not be significant, and vice versa. Therefore, we can set a threshold (e.g., one or two times of the RSME value of the fitted model). When the difference is below the threshold, the previously fitted model in (10) is still valid and no scene change occurs. Otherwise, we would update the model parameters in (10).

In the following of this section, we set the average received SNR of the Rayleigh fading channel to 10 dB and analyze accordingly the performance of the proposed dRDO-based rate

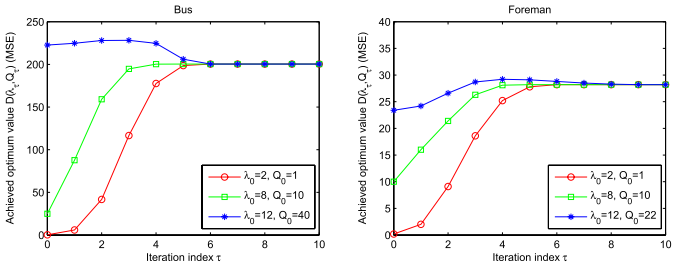


Fig. 3. Convergence behavior of the optimal objective value $D(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau)$ for (a) *Bus* sequence and (b) *Foreman* sequence.

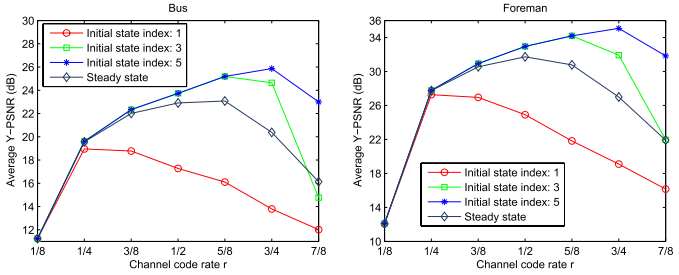


Fig. 4. Impact of different channel code rates $r$, where (a) $\Delta T_{\max} - d_c = 0.6$ s, $R_c = 100$ kb/s and $T_f = 0.2$ s for the *Bus* sequence and (b) $\Delta T_{\max} - d_c = 1$ s, $R_c = 100$ kb/s, and $T_f = 0.3$ s for the *Foreman* sequence.

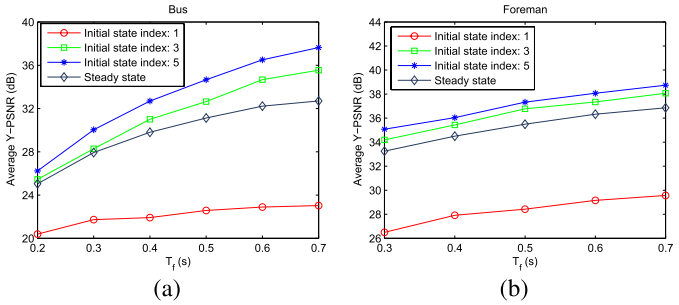


Fig. 5. Impact of different frame intervals $T_f$, where $\Delta T_{\max} - d_c = 1$ s, and $R_c = 100$ kb/s for (a) *Bus* sequence and (b) *Foreman* sequence.

control algorithm under different end-to-end system settings. To better understand the computational complexity of the proposed dRDO-based rate control algorithm, Fig. 3 shows the convergence behavior of the optimal objective value $D(\boldsymbol{\lambda}_\tau, \mathbf{Q}_\tau)$ in the subproblem (21) under different selections of initial points $\boldsymbol{\lambda}_0 = (\lambda_0, \ldots, \lambda_0)$ and $\mathbf{Q}_0 = (Q_0, \ldots, Q_0)$, for the first GOP of the *Bus* and *Foreman* video sequences. It can be seen that for a given channel code rate $r = r' \in \mathcal{R}$, the proposed rate control problem can quickly converge to the optimal solution of subproblem (21) in a few iterations, e.g., five iterations for both sequences. Since $|\mathcal{R}| = 7$, only $7 \times 5 = 35$ iterations are needed to solve Algorithm 1. Within each iteration, on the other hand, it is only required to solve a quadratic programming optimization problem. In comparison, with the typical bisection search algorithm used in [5], all the feasible regions of dual variables $\boldsymbol{\mu}$, $\boldsymbol{\eta}$, and $\boldsymbol{\varphi}$ need an iterative bisection search that greatly increases the number of iterations. Furthermore, within each iteration of the bisection search algorithm, the entire feasible sets of primal variables $\boldsymbol{\lambda} = (\lambda^1, \ldots, \lambda^k, \ldots, \lambda^K)$ and $\mathbf{Q} = (Q^1, \ldots, Q^k, \ldots, Q^K)$ are exhaustively searched to find the optimal solution, which means the duration of one iteration is much longer than that of the PA. For example, it is observed from the experiments [9] that when the upper-bound of feasible ranges for dual variables is set to 50, 13 iterations are required for convergence of the subproblem (21), and thus $7 \times 13 = 91$ total iterations for the global optimal solution with bisection search algorithm. In addition, such a number of iterations for convergence would become larger when the upper-bound increases. Therefore, the computational complexity of the PA is much lower than that of the bisection search algorithm.

In Fig. 4, we loop for all possible values of $r$ within the discrete set $\mathcal{R}$, and show the impact of different channel code rate on the average end-to-end distortion [measured in luminance-component peak signal-to-noise ratio (Y-PSNR)] for the first GOP of *Bus* and *Foreman* video sequences, under different setups of the initial channel state. It can be seen that for a given initial channel state, there is an optimal channel code rate value $r^*$ corresponding to the maximum average Y-PSNR. Though the PEP and the transmission distortion can be reduced when $r < r^*$, the available source coding rate is limited and thus the source coding distortion is increased, which will cause a larger end-to-end distortion. On the other hand, if we let $r > r^*$ to promise sufficient source coding rate and thus smaller source coding distortion, the PEP as well as the transmission incurred distortion will be increased, which will also lead to a larger end-to-end distortion. Furthermore, according to Table I, the channel state index of FSMC is ordered with the decreasing symbol error probabilities. A smaller channel state index therefore indicates a worse channel condition with higher PEP. When the initial channel state index decreases from 5 to 1, the respective initial channel condition becomes worse, and thus the optimal $r^*$ value will accordingly decrease to introduce more parity symbols for FEC. It should also be noted that the average end-to-end Y-PSNR for a given $r$ generally drops as the initial channel condition gets worse. However, for small channel code rate values (e.g., $r = 1/8$ or $1/4$), the achievable end-to-end Y-PSNR would remain similar, since at this time, the channel code rate is small enough to ensure that any transmission incurred error under different initial channel conditions can be corrected by the RS block code and thus no additional transmission distortion is introduced. In addition to the three curves studying the effect of the transient behavior of the fading channel, the steady-state performance is also shown in Fig. 4 with the corresponding curve located in the middle of these three curves.

From (19), it is observed that frame interval $T_f$, end-to-end delay bound $\Delta T_{\max} - d_c$, and channel capacity $R_c$ are the three system parameters that affect the overall performance of the end-to-end wireless video communication system. In Figs. 5–7, we separately evaluate their impacts on the average end-to-end distortion achieved by the PA for the first GOP of *Bus* and *Foreman* video sequences, under different setups of the initial channel state. For given channel capacity $R_c$ and end-to-end delay bound $\Delta T_{\max} - d_c$, as shown in

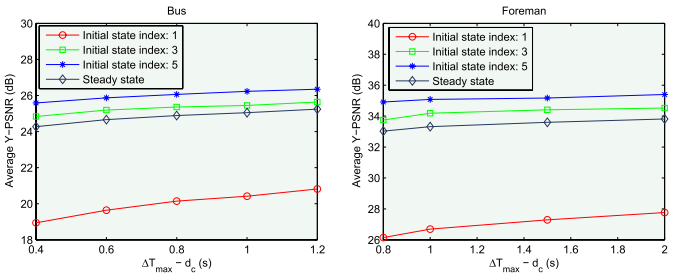


Fig. 6. Impact of different end-to-end delay bounds, where (a) $R_c = 100$ kb/s and $T_f = 0.2$ s for the *Bus* sequence and (b) $R_c = 100$ kb/s and $T_f = 0.3$ s for the *Foreman* sequence.

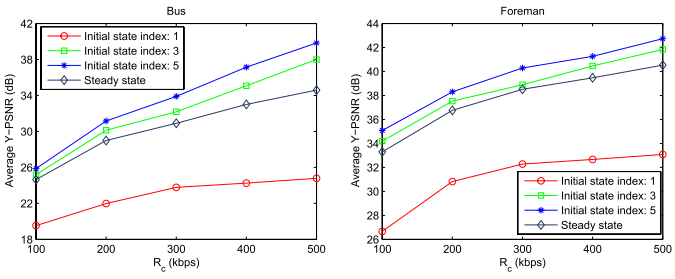


Fig. 7. Impact of different channel capacities $R_c$, where (a) $\Delta T_{\max} - d_c = 0.6$ s and $T_f = 0.2$ s for the *Bus* sequence and (b) $\Delta T_{\max} - d_c = 1$ s, and $T_f = 0.3$ s for the *Foreman* sequence.

Fig. 5, the average end-to-end Y-PSNR achieved by the PA for *Bus* and *Foreman* video sequences increases with the increment of the frame interval $T_f$. The reason is twofolded. First, a greater frame interval indicates a larger encoding delay bound for each video frame with which the source coding R-D performance could be enhanced according to [9]. Second, for a constant channel capacity, a larger frame interval will tend to result in more encoded bits allocated for each frame and thus a higher video compression quality.

Fig. 6 shows the relationship between the end-to-end delay bound $\Delta T_{\max} - d_c$ and the average end-to-end Y-PSNR, at fixed frame interval $T_f$ and channel capacity $R_c$. It can be seen that the average end-to-end Y-PSNR is an increasing function of the end-to-end delay bound. The reason is that as the end-to-end delay bound enlarges, the probability of packet drops incurred by delay bound violation would decrease accordingly. In addition, with the increment of the end-to-end delay bound, a larger number of video frames can be stored in either the encoder or the decoder buffer to mitigate the effect of instantaneous packet loss in the wireless channel, which in turn enhances the overall video reception quality.

In Fig. 7, we present the curves of the average end-to-end Y-PSNR versus channel capacity $R_c$, for the given frame interval $T_f$ and end-to-end delay bound $\Delta T_{\max} - d_c$. The average end-to-end Y-PSNR also enhances with the increment of the channel capacity $R_c$, since a larger channel transmission rate indicates that more available source coding rate can be supported by the wireless channel for each frame, and lower video compression distortion can be achieved accordingly.

### *B. Performance Comparison*

In this section, the performance of the PA is compared with four baseline schemes: 1) JM18.2 rate control algorithm [35];

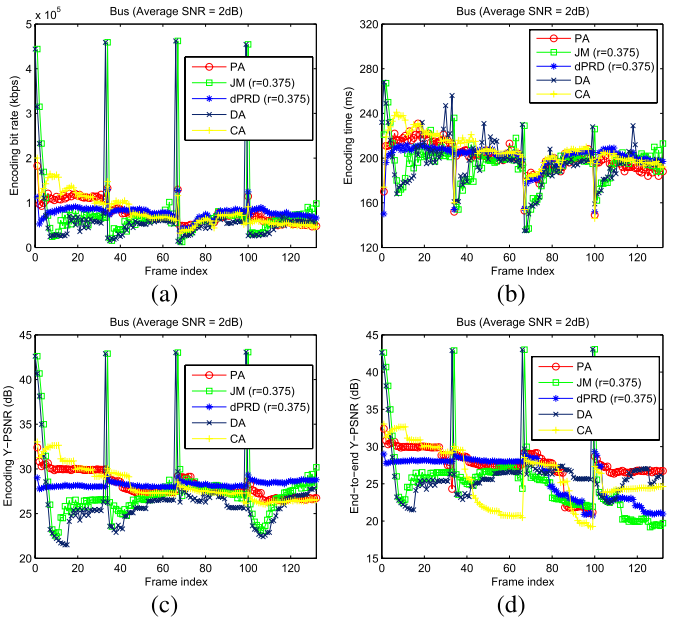


Fig. 8. Frame-wise objective quality comparison on (a) encoding bit rate, (b) encoding time, (c) encoding Y-PSNR, and (d) end-to-end Y-PSNR of different algorithms for the *Bus* sequence.

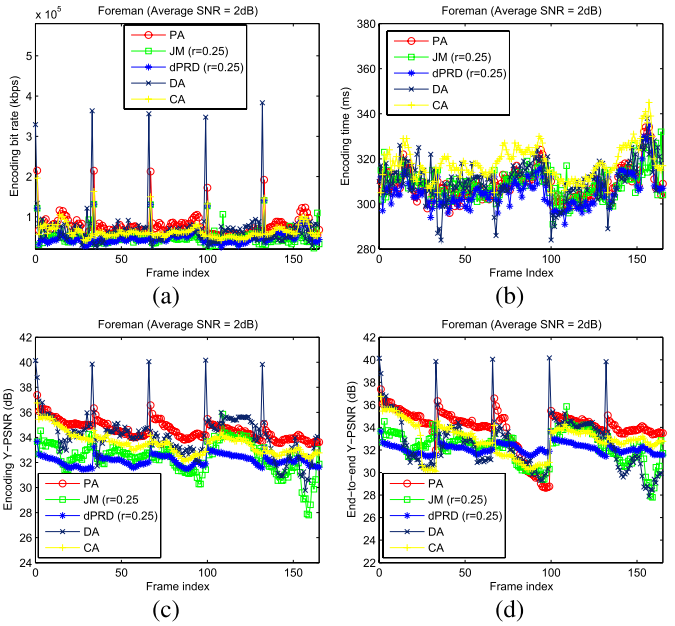


Fig. 9. Frame-wise objective quality comparison on (a) encoding bit rate, (b) encoding time, (c) encoding Y-PSNR, and (d) end-to-end Y-PSNR of different algorithms for the *Foreman* sequence.

2) delay–power–rate–distortion (dPRD) optimization-based rate control algorithm [9]; 3) delay-sensitive video streaming rate control algorithm (DA) [7]; and 4) cross-layer optimized rate control algorithm (CA) [5] that achieves the packet-level RDO determination of the encoding parameter (i.e., quantization step size) and the channel coding parameter (i.e., channel code rate) for the wireless video communication. In Figs. 8 and 9, we set the average received SNR of the Rayleigh fading channel to 2 dB, and show the frame-wise objective quality comparison of the five algorithms on encoding bit rate, encoding time, encoding Y-PSNR,

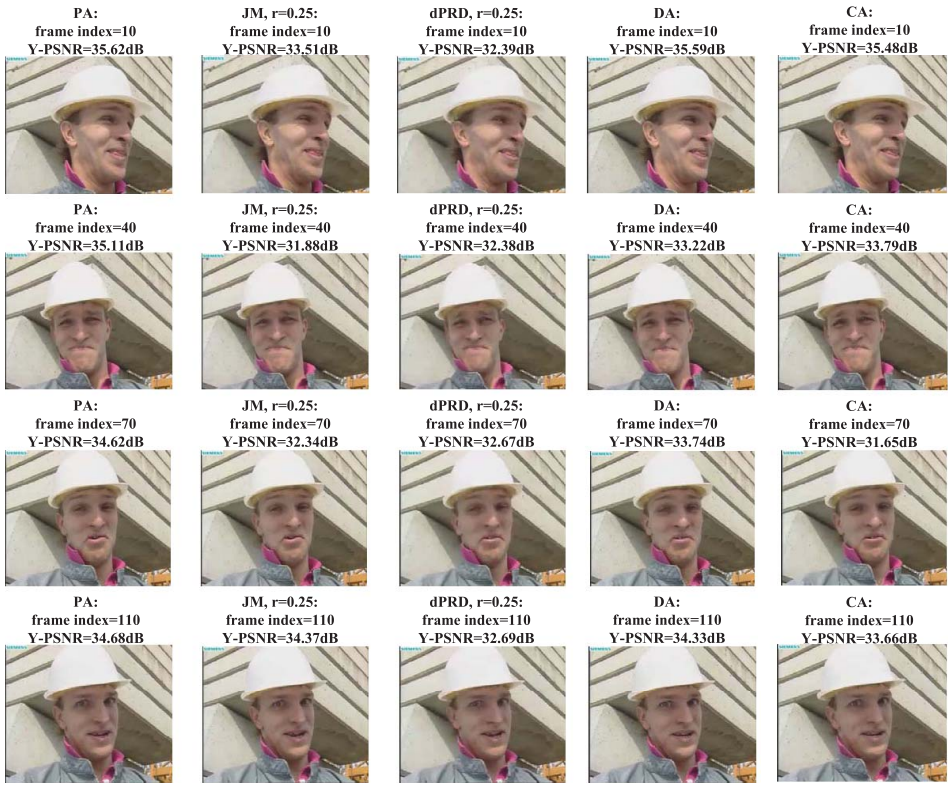


Fig. 10. Subjective quality comparison for the *Foreman* sequence, where $\Delta T_{\max} - d_c = 1$ s, $R_c = 200$ kb/s, $T_f = 0.3$ s, and average SNR = 2 dB.

and end-to-end Y-PSNR, respectively. For *Bus* sequence, the end-to-end system parameters are set as $T_f = 0.2$ s, $\Delta T_{\max} - d_c = 0.6$ s, $R_c = 200$ kb/s. Note that one of our main contributions is to determine the optimal channel code rate of the RS block code through the PA and thus to achieve the best tradeoff between the available source coding rate versus redundant rate incurred by channel coding, while the JM and dPRD schemes fail to do so. Specifically, for each GOP, the PA is able to dynamically assign an optimal channel code rate in accordance with the time-varying channel condition. However, to have a fair comparison with the PA, here we adopt RS block code as the FEC scheme for JM, dPRD, and CA, and further set the channel code rate $r$ for both JM and dPRD to the average optimal $r^*$ value of all GOPs obtained by the PA. Take the *Bus* sequence for example and $r = 3/8$ for both JM and dPRD algorithms. It should also be noted that since JM, DA, and CA all assume the encoding time of each frame to be a constant value and cannot adjust such an encoding time to meet the maximum encoding delay constraint, we further set the search range for each frame of JM, DA, and CA to the optimal value obtained by the PA.

From Fig. 8(a) and (c), it can be observed that within a specific GOP, a larger encoding bit rate generally results in a higher encoding Y-PSNR, which indicates that a larger channel code rate is assigned to that GOP to increase available source coding rate. As mentioned above, the PA can dynamically determine the optimal channel code rate for each GOP based on the corresponding channel condition. Therefore, the PA would assign a larger $r^*$ when the PEP is relatively small (e.g., frames 1–33 within the first GOP). For larger PEP, the PA could accordingly decrease the value of $r*$ to introduce more parity symbols (e.g., frames 100–132 within the fourth GOP). At this time, the achieved encoding Y-PSNR after source coding of the PA might not be the highest. However, when these encoded packets are transmitted through the error control channel and decoded by the video decoder, the overall end-to-end Y-PSNR of the PA outperforms those of the other four schemes. For example, although the encoding Y-PSNR for the fourth GOP achieved by JM, dPRD, and DA schemes might be higher with more encoded bit rate, the channel code rate is not sufficiently small to correct the transmission error introduced by such increment of source coding bit rate, which will cause greater transmission distortion and thus poorer end-to-end distortion. In general, it can be observed from Fig. 8(d) that for most of the frames, the end-to-end Y-PSNR of the PA is higher than the other four schemes. For

TABLE II
COMPARISON OF ACTUAL COMPUTATION TIME

| Sequence | Alg. | Optimization Time (s) | Encoding Time (s) | Computation Time per Frame (s) |
|---|---|---|---|---|
| Bus | PA | 5.80 | 22.56 | 0.21 |
| | JM | 0 | 21.08 | 0.16 |
| | dPRD | 1.23 | 22.48 | 0.18 |
| | DA | 0.06 | 21.15 | 0.16 |
| | CA | 7.05 | 22.26 | 0.22 |
| Foreman | PA | 7.27 | 46.89 | 0.33 |
| | JM | 0 | 45.96 | 0.28 |
| | dPRD | 1.72 | 46.20 | 0.29 |
| | DA | 0.23 | 45.96 | 0.28 |
| | CA | 8.76 | 46.35 | 0.33 |

TABLE III
COMPARISON OF AVERAGE Y-PSNR (DECIBELS) UNDER DIFFERENT AVERAGE SNR AND CHANNEL CAPACITIES

| Sequence | Average SNR (dB) | $R_c$ (kbps) | PA | JM | | | dPRD | | | DA | CA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Bus | 2 | | | r=0.25 | r=0.375 | r=0.5 | r=0.25 | r=0.375 | r=0.5 | | |
| | | 100 | 24.13 | 22.72 | 22.58 | 19.72 | 22.91 | 22.84 | 20.07 | 22.97 | 23.60 |
| | | 200 | 27.36 | 24.56 | 25.35 | 21.11 | 25.86 | 25.89 | 21.97 | 26.79 | 26.65 |
| | | 500 | 33.40 | 28.79 | 29.00 | 24.15 | 30.69 | 29.19 | 24.77 | 32.77 | 32.59 |
| | 5 | | | r=0.375 | r=0.5 | r=0.625 | r=0.375 | r=0.5 | r=0.625 | | |
| | | 100 | 26.36 | 23.62 | 24.56 | 22.41 | 24.40 | 25.72 | 22.65 | 24.03 | 25.50 |
| | | 200 | 30.47 | 27.43 | 27.63 | 24.20 | 28.27 | 29.91 | 24.62 | 27.52 | 29.38 |
| | | 500 | 37.49 | 33.36 | 36.26 | 27.71 | 33.59 | 36.52 | 28.13 | 36.50 | 36.37 |
| | 10 | | | r=0.625 | r=0.75 | r=0.875 | r=0.625 | r=0.75 | r=0.875 | | |
| | | 100 | 28.47 | 26.13 | 27.24 | 19.00 | 26.82 | 27.90 | 18.09 | 26.71 | 27.64 |
| | | 200 | 32.71 | 30.28 | 32.03 | 19.49 | 30.95 | 32.16 | 18.69 | 31.59 | 31.97 |
| | | 500 | 42.52 | 39.92 | 41.65 | 19.89 | 40.65 | 41.84 | 19.77 | 42.15 | 41.60 |
| Foreman | 2 | | | r=0.25 | r=0.375 | r=0.5 | r=0.25 | r=0.375 | r=0.5 | | |
| | | 100 | 30.87 | 28.65 | 29.29 | 25.65 | 28.17 | 29.36 | 25.71 | 29.47 | 30.07 |
| | | 200 | 33.79 | 32.32 | 31.88 | 26.95 | 32.13 | 31.89 | 26.94 | 32.82 | 33.14 |
| | | 500 | 37.59 | 36.98 | 34.01 | 28.33 | 36.80 | 34.05 | 28.11 | 37.02 | 36.91 |
| | 5 | | | r=0.375 | r=0.5 | r=0.625 | r=0.375 | r=0.5 | r=0.625 | | |
| | | 100 | 33.15 | 30.95 | 32.32 | 28.96 | 31.10 | 32.03 | 28.74 | 32.05 | 32.22 |
| | | 200 | 36.38 | 34.58 | 35.68 | 30.29 | 34.91 | 35.71 | 30.08 | 35.82 | 35.40 |
| | | 500 | 40.51 | 39.02 | 39.78 | 31.79 | 38.88 | 39.74 | 31.57 | 40.30 | 39.50 |
| | 10 | | | r=0.625 | r=0.75 | r=0.875 | r=0.625 | r=0.75 | r=0.875 | | |
| | | 100 | 35.10 | 33.50 | 34.39 | 24.37 | 34.08 | 34.52 | 24.28 | 34.21 | 33.57 |
| | | 200 | 38.21 | 37.15 | 37.64 | 24.75 | 37.50 | 37.60 | 24.62 | 37.84 | 37.69 |
| | | 500 | 42.50 | 41.69 | 41.81 | 25.20 | 41.78 | 41.70 | 25.00 | 41.96 | 41.85 |
| Mobile | 2 | | | r=0.25 | r=0.375 | r=0.5 | r=0.25 | r=0.375 | r=0.5 | | |
| | | 100 | 20.34 | 18.44 | 19.77 | 18.27 | 19.01 | 19.80 | 17.98 | 19.58 | 19.85 |
| | | 200 | 22.96 | 22.33 | 21.92 | 18.98 | 21.36 | 21.37 | 18.85 | 22.51 | 22.41 |
| | | 500 | 25.98 | 24.70 | 24.34 | 20.58 | 24.89 | 24.29 | 20.25 | 24.87 | 25.77 |
| | 5 | | | r=0.375 | r=0.5 | r=0.625 | r=0.375 | r=0.5 | r=0.625 | | |
| | | 100 | 21.84 | 20.41 | 21.29 | 20.25 | 20.90 | 21.35 | 19.93 | 21.51 | 21.34 |
| | | 200 | 24.18 | 23.00 | 23.77 | 21.52 | 22.61 | 23.73 | 21.20 | 23.67 | 23.44 |
| | | 500 | 28.85 | 26.63 | 28.10 | 23.99 | 26.75 | 28.14 | 23.82 | 28.15 | 28.26 |
| | 10 | | | r=0.625 | r=0.75 | r=0.875 | r=0.625 | r=0.75 | r=0.875 | | |
| | | 100 | 23.29 | 22.56 | 22.96 | 17.79 | 22.11 | 22.54 | 17.37 | 22.99 | 22.63 |
| | | 200 | 25.75 | 24.74 | 25.35 | 18.18 | 24.90 | 25.33 | 17.89 | 25.30 | 25.38 |
| | | 500 | 32.56 | 30.13 | 31.45 | 19.02 | 30.28 | 31.91 | 18.89 | 31.64 | 31.26 |

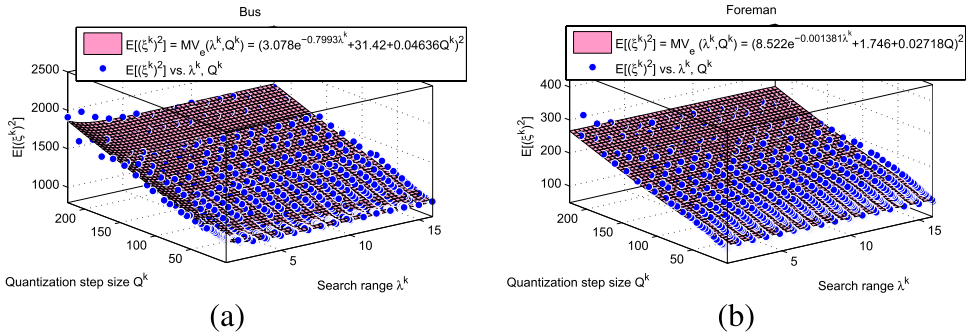


Fig. 11. 2-D fitting of $E[(\xi^k)^2]$ versus search range and quantization step size. (a) *Bus* video sequence with R-square = 0.973 and RMSE = 31.75. (b) *Foreman* video sequence with R-square = 0.9314 and RMSE = 10.7.

the encoding delay, as shown in Fig. 8(b), the single frame delay constraint is approximately satisfied by all schemes. The similar result is also shown in Fig. 9, where we let $T_f = 0.3$ s, $\Delta T_{\max} - d_c = 1$ s, and $R_c = 200$ kb/s for the *Foreman* sequence.

In Table II, we compare the actual computation time that is spent by different algorithms to obtain the results shown in Figs. 8 and 9. Theoretically, the computation time of each algorithm includes both the optimization time spent in the iterative selection of the optimal parameters and the total encoding time used to encode the sequence. Accordingly, Table II shows these two types of time for each algorithm operation in practice, and the computation time per frame is obtained by averaging the sum of the optimization time and the total encoding time over all the frames. Since the RDO in the JM rate control scheme is integrated in the encoder and utilized for encoding each frame by tuning respective $QP$ to meet the target bit rate while minimizing the coding distortion, no additional optimization time is introduced. In comparison, the other four algorithms need to determine the optimal coding parameters before encoding the frames, which will cause additional optimization time apart from the encoding time. It can be seen that for each sequence, CA is the most time consuming in optimization iteration

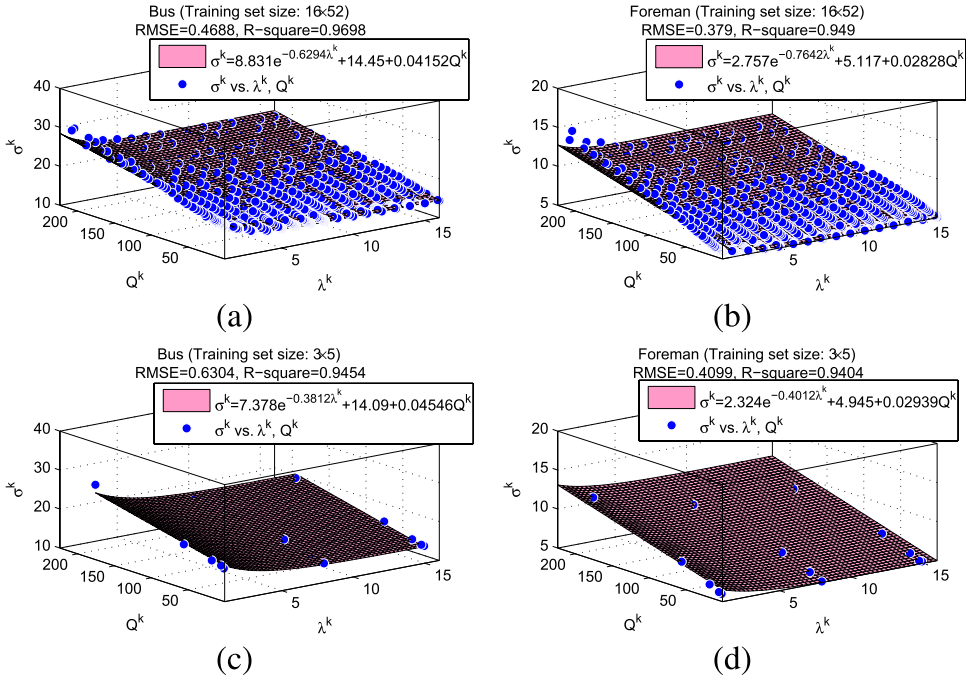


Fig. 12. Fitting result of (10) based on different training set. (a) and (b) Whole set $S_0 = \{\lambda^k, QP^k | 1 \le \lambda^k \le 16, 0 \le QP^k \le 51\}$ and (c) and (d) subset $S_1 = \{\lambda^k, QP^k | \lambda^k = 7 \times i + 1, QP^k = 10 \times j + 9, i = 0, 1, 2, j = 0, 1, \ldots, 4\}$ with three different $\lambda^k$ values and five different $QP^k$ values.

since it adopts iterative bisection search to find the optimal quantization step size and channel coding rate (note that CA cannot select the optimal search range). As shown in Table II, the additional optimization time introduced by the PA is several seconds. If such optimization time is averaged over all the frames contained in the sequence, its impact becomes trivial. For example, the computation time (including the optimization time and the encoding time) per frame of the PA is 0.21 and 0.33 s for the *Bus* and *Foreman* sequences, respectively, which is comparable with the frame interval of these two sequences ($T_f = 0.2$ s for the *Bus* sequence and $T_f = 0.3$ s for the *Foreman* sequence).

In addition, to give a visual comparison, the subjective quality of the five different algorithms for the *Foreman* video sequence is shown in Fig. 10, with end-to-end Y-PSNR values attached. It can also be seen that the subjective quality of decoded frames of the PA outperforms those of the other four algorithms. It should be noted that the difference among frames constructed by different algorithms mainly exists in the texture regions with detailed information, e.g., the hair and collar of the *Foreman* in Fig. 10.

Table III shows the comparison of the five different algorithms on average end-to-end Y-PSNR versus average received SNR and channel capacity of the Rayleigh fading channel, for the *Bus*, *Foreman*, and *Mobile* video sequences. Similarly, with the PA, the optimal channel code rate $r^*$ is solved by (19) for each specific GOP. Since JM and dPRD schemes can not determine such optimal value, we set their corresponding $r$ to three values around the average $r^*$ of all GOPs to have a fair comparison. It can be observed from Table III that for the same average received SNR, all the five algorithms will achieve higher average end-to-end Y-PSNR as the channel capacity increases, which allows more source coding bit rate to be supported by the wireless channel. On the other hand, when the channel capacity is fixed, the average end-to-end Y-PSNR for all the four algorithms would become higher with the increment of the average received SNR, since a larger average SNR corresponds to a better channel condition and thus a smaller PEP. In general, for the given average received SNR and channel capacity, the average Y-PSNR of the PA achieved by the optimal $r^*$ values for multiple GOPs is higher than those of the other four schemes.

## VI. Conclusion

We developed for the end-to-end wireless video communication system a dRDO rate control problem to minimize the end-to-end distortion subject to the transmission rate and end-to-end delay constraints. The tradeoffs regarding source coding delay versus buffering delay and available source coding rate versus redundant rate incurred by channel coding were coupled in the proposed problem. To solve it, we proposed a practical algorithm using the Lagrange multiplier approach, the KKT conditions, and the SQP methods. The experimental results have verified the optimization performance of the PA. Our future work will focus on studying the joint source and channel resource allocation over multihop networks by exploring WiMAX and LTE.

## Appendix A
## Derivation of $E[(\xi^k)^2]$

In a similar way to the derivation process of $\sigma^k(\lambda^k, Q^k)$, as introduced in [9], $E[(\xi^k)^2]$ can be fitted as a closed-form function of search range and quantization step size,

i.e., $MV_e(\lambda^k, Q^k)$. Accordingly, Fig. 11 shows the 2-D fitting results of the function $MV_e(\lambda^k, Q^k)$, with both $R$-square and root mean square error (RMSE) metrics used to measure the fitting accuracy.

## APPENDIX B
## DERIVATION OF $D(\boldsymbol{\lambda}, \mathbf{Q})$

From (12), we have

$$D_t^k(\lambda^k, Q^k, r) \tag{28a}$$

$$\triangleq \bar{P}^k \cdot \{[\sigma^k(\lambda^k, Q^k)]^2 + \rho^k \cdot MV_e(\lambda^k, Q^k)\} + z^k \cdot D_t^{k-1} \tag{28b}$$

$$= \cdots$$

$$= \sum_{i=1}^{k} \left( \prod_{j=i+1}^{k} z^j \right) \cdot \bar{P}^i \cdot \left\{[\sigma^i(\lambda^i, Q^i)]^2 + \rho^i \cdot MV_e(\lambda^i, Q^i)\right\}$$

$$\approx \sum_{i=1}^{k} \bar{P}^i \cdot \left\{[\sigma^i(\lambda^i, Q^i)]^2 + \rho^i \cdot MV_e(\lambda^i, Q^i)\right\} \tag{28c}$$

where (28b) is obtained by defining $z^k = \bar{P}^k + (1 - \bar{P}^k) \cdot \alpha^k$ and (28c) is approximately derived through approximating the propagation factor $\alpha^k$ as 1. Integrating (28) into (20a), the optimization objective of the subproblem (20) can be reformulated as

$$\begin{aligned}
&D(\boldsymbol{\lambda}, \mathbf{Q}) \\
&\triangleq \frac{1}{K} \sum_{k=1}^{K} \left[D_e^k(\lambda^k, Q^k) + D_t^k(\lambda^k, Q^k, r)\right] \\
&= \frac{1}{K} \left\{ \sum_{k=1}^{K} D_e^k(\lambda^k, Q^k) + \sum_{k=1}^{K} D_t^k(\lambda^k, Q^k, r) \right\} \\
&= \frac{1}{K} \left\{ \sum_{k=1}^{K} D_e^k(\lambda^k, Q^k) \right. \\
&\qquad \left. + \sum_{k=1}^{K} \sum_{i=1}^{k} \bar{P}^i \{[\sigma^i(\lambda^i, Q^i)]^2 + \rho^i MV_e(\lambda^i, Q^i)\} \right\} \\
&= \frac{1}{K} \sum_{k=1}^{K} D_e^k(\lambda^k, Q^k) \\
&\quad + \frac{1}{K} \left\{ \sum_{k=1}^{K} \sum_{i=1}^{k} \bar{P}^i \{[\sigma^i(\lambda^i, Q^i)]^2 + \rho^i MV_e(\lambda^i, Q^i)\} \right\} \\
&= \frac{1}{K} \sum_{k=1}^{K} D_e^k(\lambda^k, Q^k) \\
&\quad + \frac{1}{K} \sum_{k=1}^{K} (K+1-k) \{\bar{P}^k \{[\sigma^k(\lambda^k, Q^k)]^2 + \rho^k MV_e(\lambda^k, Q^k)\}\}.
\end{aligned} \tag{29}$$

## APPENDIX C
## TRAINING SET SELECTION OF (10)

To determine the proper subset for (10), we can select a much smaller number of $\lambda^k$ and $Q^k$ values (or quantization parameter $QP^k$) which are near-uniformly distributed over the ranges of $\lambda^k$ and $Q^k$. As for justification, in Fig. 12, we reduce the training set size and study the accuracy of the fitted model in (10). As the size of the training set reduces, the accuracy of the fitted model in (10) would only slightly decrease. Therefore, the $3 \times 5$ subset $S_1$ can be viewed as a proper subset of $\lambda^k$ and $Q^k$ to obtain the model parameters in (10) with fitting accuracy comparable with the whole set $S_0$. However, the computational complexity is significantly reduced, since only $3 \times 5 = 15$ pairs of $(\lambda^k, Q^k)$ are used.

## REFERENCES


[1] Cisco Visual Networking Index (VNI) Forecast. (2014). *Global Mobile Data Traffic Forecast Update, 2013-2018*. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.pdf

[2] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1012–1032, Jun. 2000.

[3] C.-Y. Hsu, A. Ortega, and M. Khansari, "Rate control for robust video transmission over burst-error wireless channels," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 5, pp. 756–773, May 1999.

[4] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 511–523, Jun. 2002.

[5] Z. Chen and D. Wu, "Rate-distortion optimized cross-layer rate control in wireless video communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 3, pp. 352–365, Mar. 2012.

[6] S. Soltani, K. Misra, and H. Radha, "Delay constraint error control protocol for real-time video communication," *IEEE Trans. Multimedia*, vol. 11, no. 4, pp. 742–751, Jun. 2009.

[7] H. Bobarshad, M. van der Schaar, A. H. Aghvami, R. S. Dilmaghani, and M. R. Shikh-Bahaei, "Analytical modeling for delay-sensitive video over WLAN," *IEEE Trans. Multimedia*, vol. 14, no. 2, pp. 401–414, Apr. 2012.

[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 2006.

[9] C. Li, D. Wu, and H. Xiong, "Delay—Power-rate-distortion model for wireless video communication under delay and energy constraints," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 7, pp. 1170–1183, Jul. 2014.

[10] Q. Chen, "Image and video processing for denoising, coding and content protection," Ph.D. dissertation, Dept. Elect. Comput. Eng., Univ. Florida, Gainesville, FL, USA, 2011.

[11] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos, "Joint source-channel coding for motion-compensated DCT-based SNR scalable video," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1043–1052, Sep. 2002.

[12] Q. Chen and D. Wu, "Delay-rate-distortion model for real-time video communication," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[13] C.-Y. Hsu, A. Ortega, and A. R. Reibman, "Joint selection of source and channel rate for VBR video transmission under ATM policing constraints," *IEEE J. Sel. Areas Commun.*, vol. 15, no. 6, pp. 1016–1028, Aug. 1997.

[14] S. Pudlewski, N. Cen, Z. Guan, and T. Melodia, "Video transmission over lossy wireless networks: A cross-layer perspective," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 1, pp. 6–21, Feb. 2015.

[15] Z. Guan, T. Melodia, and D. Yuan, "Jointly optimal rate control and relay selection for cooperative wireless video streaming," *IEEE/ACM Trans. Netw.*, vol. 21, no. 4, pp. 1173–1186, Aug. 2013.

[16] C. Gong and X. Wang, "Adaptive transmission for delay-constrained wireless video," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 49–61, Jan. 2014.

[17] Z. Chen and D. Wu, "Prediction of transmission distortion for wireless video communication: Analysis," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1123–1137, Mar. 2012.

[18] Z. Chen and D. Wu, "Prediction of transmission distortion for wireless video communication: Algorithm and application," *J. Vis. Commun. Image Represent.*, vol. 21, no. 8, pp. 948–964, Nov. 2010.

[19] Z. He, Y. Liang, L. Chen, I. Ahmad, and D. Wu, "Power-rate-distortion analysis for wireless video communication under energy constraints," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 5, pp. 645–658, May 2005.
[20] M. J. Neely, E. Modiano, and C. E. Rohrs, "Dynamic power allocation and routing for time-varying wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 89–103, Jan. 2005.
[21] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Laplace distribution based Lagrangian rate distortion optimization for hybrid video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 2, pp. 193–205, Feb. 2009.
[22] H.-M. Hang and J.-J. Chen, "Source model for transform video coder and its application. I. Fundamental theory," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 287–298, Apr. 1997.
[23] H. S. Wang and N. Moayeri, "Finite-state Markov channel—A useful model for radio communication channels," *IEEE Trans. Veh. Technol.*, vol. 44, no. 1, pp. 163–171, Feb. 1995.
[24] Q. Zhang and S. A. Kassam, "Finite-state Markov model for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1688–1692, Nov. 1999.
[25] J. Chen, V. K. N. Lau, and Y. Cheng, "Distributive network utility maximization over time-varying fading channels," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2395–2404, May 2011.
[26] H. Cui, C. Luo, C. W. Chen, and F. Wu, "Robust linear video transmission over Rayleigh fading channel," *IEEE Trans. Commun.*, vol. 62, no. 8, pp. 2790–2801, Aug. 2014.
[27] H. Kim, P. C. Cosman, and L. B. Milstein, "Motion-compensated scalable video transmission over MIMO wireless channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 1, pp. 116–127, Jan. 2013.
[28] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Oper. Res.*, vol. 11, no. 3, pp. 399–417, May/Jun. 1963.
[29] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Advanced Lagrange multiplier selection for hybrid video coding," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 364–367.
[30] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers [speech coding]," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 9, pp. 1445–1453, Sep. 1988.
[31] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 23–50, Nov. 1998.
[32] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.
[33] B. Chachuat, *Nonlinear and Dynamic Optimization: From Theory to Practice*. Lausanne, Switzerland: École Polytechnique Fédérale de Lausanne, 2007.
[34] J. Nocedal and S. Wright, *Numerical Optimization*. New York, NY, USA: Springer-Verlag, 2006.
[35] K. Sühring. (Feb. 2012). *H.264/AVC Reference Software JM18.2*. [Online]. Available: http://iphome.hhi.de/suehring/tml/download/
[36] D. Lelescu and D. Schonfeld, "Statistical sequential analysis for real-time video scene change detection on compressed multimedia bitstream," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 106–117, Mar. 2003.

**Chenglin Li** (S'12–M'15) received the B.S., M.S., and Ph.D. degrees in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 2007, 2009, and 2015, respectively.

He was with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA, as a Visiting Ph.D. Student. His research interests include network oriented image/video processing and communication, and the network-based optimization for video sources.

**Hongkai Xiong** (M'01–SM'10) received the Ph.D. degree in communication and information system from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2003.

He was with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, from 2007 to 2008, as a Research Scholar. From 2011 to 2012, he was a Scientist with the Division of Biomedical Informatics, University of California at San Diego, La Jolla, CA, USA. He has been with the Department of Electronic Engineering, SJTU, where he is currently a Full Professor. He has authored over 130 refereed journal/conference papers. His research interests include source coding/network information theory, signal processing, computer vision, and machine learning.

Dr. Xiong received the Best Student Paper Award at the 2014 IEEE Visual Communication and Image Processing in 2014, the Best Paper Award at the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting in 2013, and the Top 10% Paper Award at the IEEE International Workshop on Multimedia Signal Processing in 2011. In 2014, he was granted the National Science Fund for Distinguished Young Scholar and the Shanghai Youth Science and Technology Talent. He received the Shanghai Shu Guang Scholar in 2013. Since 2012, he has been a member of the Innovative Research Group of the National Natural Science. He also received the First Prize of the Shanghai Technological Innovation Award for Network-Oriented Video Processing and Dissemination: Theory and Technology in 2011, and the SMC-A Excellent Young Faculty Award of Shanghai Jiao Tong University in 2010 and 2013. In 2009, he was a recipient of the New Century Excellent Talents in University, Ministry of Education of China. He served as a Technical Program Committee Member for prestigious conferences, such as the ACM Multimedia, the International Conference on Image Processing, the International Conference on Multimedia and Expo, and the International Symposium on Circuits and Systems.

**Dapeng Wu** (S'98–M'04–SM'06–F'13) received the B.E. degree in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 1990; the M.E. degree in electrical engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 1997; and the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2003.

He is a Professor with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA. His research interests include networking, communications, signal processing, computer vision, machine learning, smart grid, and information and network security.

Dr. Wu received the University of Florida Research Foundation Professorship Award in 2009, the AFOSR Young Investigator Program (YIP) Award in 2009, the ONR YIP Award in 2008, the NSF CAREER Award in 2007, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Best Paper Award in 2001, and the best paper awards in IEEE GLOBECOM in 2011 and the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks in 2006. He is an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS. He is the Founder of the IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING. He was the Founding Editor-in-Chief of *Journal of Advances in Multimedia* from 2006 to 2008 and an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY from 2004 to 2007. He is a Guest Editor of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS of the Special Issue on Cross-Layer Optimized Wireless Multimedia Communications. He was a Technical Program Committee Chair of the IEEE INFOCOM in 2012, the IEEE International Conference on Communications in 2008, and the Signal Processing for Communications Symposium, and a member of the Executive Committee and Technical Program Committee of over 80 conferences. He was the Chair of the Award Committee, the Mobile and Wireless Multimedia Interest Group, the Technical Committee on Multimedia Communications, and the IEEE Communications Society. He was an elected member of the Multimedia Signal Processing Technical Committee and the IEEE Signal Processing Society from 2009 to 2012.