Optimal Packet Scheduling for Multi-Description Multi-Path Video Streaming Over Wireless Networks

Gui Xie, Member, IEEE, M.N.S Swamy, Fellow, IEEE and M. Omair Ahmad, Fellow, IEEE
Department of Electrical and Computer Engineering
Concordia University
Montreal, Canada, H3G 1M8
Email: guixie@encs.concordia.ca

Abstract—As developments in wireless networks continue, there is an increasing expectation with regard to supporting high-quality real-time video streaming service in such networks. The recent advances in multi-description (MD) multi-path transport has made it a promising technology for content-rich wireless multimedia communications. This paper presents a rate-distortion (R-D) optimized packet scheduling algorithm (OPT-MD) for streaming MD-coded video along multiple wireless paths. Our algorithm relies on R-D hint information that is used to characterize a packet in a R-D sense. The information consists of the size of the packet in bits and the importance of the packet for reconstructing the video. Each of the video description adaptively selects certain important packets for transmission according to the quality of the transmission path by simultaneously considering bandwidth, bit error rate, and delay so that the overall end-to-end video distortion in terms of the mean square error (MSE) is minimized. Extensive simulation results demonstrate that OPT-MD can improve the quality of video streaming significantly as compared to a conventional scheduling approach that does not consider the relative importance of the video packets and the channel conditions (RANDOM-MD). The gains in performance reach up to 5 dB and 4 dB for streaming MD-coded format QCIF FORMAN and TABLE video sequences, respectively, in the scenario of adaptation to a simulated time-varying network channel. Our efforts in this work provides an important methodology for high-quality real-time video streaming applications over wireless networks.

I. INTRODUCTION

Realtime video streaming over wireless networks has attracted many research efforts over the past few years [1]. However, the nature of wireless multi-hop networks, lack of a pre-installed infrastructure, potentially-low bandwidth, high bit error rates and stringent delay constraints, makes it a great challenge to offer connections of a quality sufficient for this media-rich application. A potentially promising approach to this problem is to establish multiple paths between the source and the destination for a video traffic session and to use coding schemes that take advantage of the existence of multiple paths. One such popular coding scheme is multi-description (MD) coding [2]. With MD coding, multiple substreams, also called descriptions, are produced for a video source, each of which is roughly of equal importance, such that any received description can be independently decoded to give an acceptable reproduction of the original video signal and the quality of the decoded signal is commensurate with additional descriptions. Each description is sent along a different path. Although most of the paths in wireless networks are highly error-prone, as long as the link failure events on different paths are not entirely correlated, the probability of concurrent loss of all of the descriptions will be low. Therefore, MD coding will remain effective for most of the streaming period, while video quality improves as more descriptions are received.

There has been considerable research on using MD coding with multi-path routing to enable video streaming over wireless networks [2]. Recently, several researchers have proposed the joint routing and encoding approach to find the optimal paths and encoding rates for the MD video in order to minimize the application layer video distortion metric, assuming that the raw video data are available at the sender and the MD-coded streams can be generated on the fly [3], [9].

As compared to the case where real-time encoding is performed, a more realistic application scenario nowadays in wireless networks, for instance, Video-on-Demand (VoD) streaming, is that the encoded video descriptions with fixed video rates are pre-stored at the source and transmitted to the destination on demand. Therefore, when the transmission bandwidth of a path maintained by the underlying network layer for a video description is insufficient or become insufficient because of the wireless channel variations, one needs to reduce the transmission rate in order to account for it. This is in turn achieved by skipping packets prior to transmission due to the timing constraints of the underlying streaming application. Now randomly skipping packets can have an unpredictable effect on the quality of the reconstruction video. Performing proper video packet selection for MD-coded video transmission in such a bandwidth-constrained and packet-lossy setting can be an involved packet scheduling task.

Solutions may be proposed that try to adapt the video representation to the channel variations, however, these solutions will be at the price of high complexity, or loss in coding performance. Video transcoding [4], for example, decodes the pre-stored video description to a raw intermediate format and re-encodes it into the target bit rate, but this requires significant
complexity and computation. Scalable coding techniques have been developed to solve these problems, where the encoding provides an inherent prioritization among the compressed data and allows the decoded data rate to be dynamically changed. This offers a natural method for selecting certain portions of the compressed data to meet the transmission bandwidth. However, scalable streams have not gained a wide acceptance due to their coding inefficiency. On the other hand, non-scalable hybrid transform and motion compensated video coding is predominantly used in streaming today. For instance, a simple and efficient MD coding technique is to split the original video sequence into even-indexed and odd-indexed frames and then compress them into two descriptions separately using the latest H.264 encoder. However, it does not suggest a natural method of placing delivery priorities on compressed video packets. Adaptive MD coded video streaming via optimal selection of non-scalable encoded video packets is the focus of our work in this paper.

We propose a rate-distortion (R-D) optimized packet scheduling algorithm for MD-coded video streaming along multiple wireless paths. Each of the individual video descriptions adaptively selects a portion of the packets for delivery according to the characteristics of its transmission path (e.g., bandwidth, delay and loss behavior) in an R-D optimized way such that the end-to-end performance in terms of video quality, averaged over all the video descriptions, is maximized. Our optimization framework relies on the R-D hint information, i.e., rate-distortion hint track (RDHT) proposed in [5], which basically consist of two quantities: the size of the packet in bits and the importance of the packet in an R-D sense. The R-D hint information can be computed at the encoding time, simply stored in the same file as the compressed video data, and de-multiplexed when necessary.

There is a substantial body of prior literature on packet selection and scheduling for video streaming, over wireless WANs and over packet-lossy networks in general. However, to the best of our knowledge, R-D optimized packet selection for MD-coded video streaming using multi-path routing over wireless networks, as studied in the present paper, has not been investigated. The most-closely related works are the following. By giving priority to perceptually more important packets at (re)transmission, a cross-layer ARQ algorithm is proposed for video streaming in 802.11 wireless networks [6]. Only a single video stream is considered. The authors in [7] design a transmission strategy that provides adaptive quality of service (QoS) to layered video for streaming over 802.11 WLANs. Again, only a single video stream is considered and no R-D optimization is performed. Similarly, hybrid transmission techniques that combine Automatic Repeat reQuest (ARQ) and Forward Error Correction (FEC) are proposed for improved real-time video transport over WLANs. The authors in [5] present two RDHT-based systems, which perform R-D optimized scheduling for a single video stream. In [8], the authors extend the optimization framework of [5] to the scenario of distributed streaming of concurrent multiple video streams sharing a single transmission path. By exploring server diversity instead of path diversity, a client-driven R-D optimized packet scheduling algorithm for streaming MD-coded video is presented in [9], where the video descriptions are distributed among the different VoD servers. Finally, our work is perhaps most closely related to [10], which propose a R-D optimized packet scheduling method for MD-coded video streaming over packet-lossy networks when only a single physical path is considered. However, the above techniques do not extend to the case of MD-coded video streaming over multiple paths, which is the scenario studied in this paper.

The rest of this paper is organized as follows. In Section II, we describe the general multi-description multi-path video streaming architecture, introduce how to compute the R-D hint information, and characterize a wireless path between the source and the destination. In Section III, we formulate the optimized scheduling problem and propose an efficient solution. In Section IV, we compare the performance of our optimized scheduling framework to that of a conventional system for MD video streaming over wireless networks. Finally, Section V contains the conclusions.

II. PRELIMINARIES

A. Multi-Description Multi-Path Transport

The general architecture of MD-coded video streaming using multi-path transport is illustrated in Figure 1, where multiple descriptions of a source video are transmitted along different paths from the source node to the destination node. We assume that an underlying multi-path routing protocol maintains these multiple paths and reports the path parameters such as bandwidth, delay and bit error rate to the application layer periodically.

On the sender side, the original video frame sequence is split to several sub-sequences (e.g., even-indexed and odd-indexed frames) and each of these sub-sequences is encoded to a video description by H.264/AVC or any other hybrid transform and motion-compensated video coders such as MPEG-1 and MPEG-2. Each video description is then fed into a packet scheduler, which is responsible for selecting and dispatching appropriate data packets onto its transmission path. The packet scheduling strategy is affected by the QoS requirements and path characteristics. Usually, the path parameters are collected from the underlying network layer periodically, so that a packet scheduler can adjust its scheduling strategy to changes in the network. On the receiver side, received packets are reassembled into video descriptions in a resequencing buffer. The MD decoder will attempt to perform video reconstruction after a preset playout delay. A packet may be lost because of transmission errors, overdue delivery, or dropped by the scheduler prior to transmission due to the bandwidth constraint. All these types of packet losses are undesirable in terms of application QoS requirements. A major concern of our packet scheduler design in the present paper is as to how to maximize the quality of the reconstructed video on the receiver side by simultaneously considering different effects of these packet losses.
by the following parameters.
- \( b_h \): the end-to-end available bandwidth
- \( e_h \): the end-to-end bit error rate
- \( t_h \): the end-to-end packet delay

In this paper, we assume that the above path parameters are time-invariant over short time scales and can vary slowly over time.

By modeling the bit errors as a Bernoulli event, the probability of losing packet \( k \) of \( V_h \) during transmitting this packet along the path \( PT_h \) can be computed as
\[
P_h(k) = 1 - (1 - e_h)^{R_h(k)}
\]

The packet delay, \( t_h \), is random with a probability density, say, \( f_h(t) \). Each packet is lost or delayed independently of the other packets. We do not assume any particular form for the density function \( f_h \). However, for specificity in the next section and in Section IV (Experimental Results) we model the packet delay as having a shifted Gamma distribution with rightward shift \( \mu_F \) and parameters \( n_F \) and \( \alpha_F \), as in [11]. The cumulative distribution function for the packet delay is
\[
P\{t_h > \Delta\} = \int_{\Delta}^{\infty} f_h(x) dx
\]
where
\[
f_h(x) = \frac{\alpha_F}{\Gamma(n_F)} (\alpha(x - \kappa_F))^{n_F - 1} e^{-\alpha_F(x-k)}
\]
for \( x \geq \kappa_F \). This distribution is equal to a constant \( \kappa_F \) plus the sum of \( n \) independent identically distributed exponential random variables, each with the parameter \( \alpha_F \). One way to interpret this is that the packet delay is the result of a packet going through \( n_F \) wireless nodes of the transmission path, each of which requires a constant processing time \( \kappa_F/n_F \) plus a waiting time in a steady M/M/1 queue with an exponential distribution \( f(x) = \alpha_F e^{-\alpha_F x^2} \). Since an exponential random variable with parameter \( \alpha \) has a mean of \( 1/\alpha \) and a variance of \( 1/\alpha^2 \), the packet delay has a mean of \( \mu_F = \kappa_F + n_F/\alpha_F \) and a variance of \( \sigma_F^2 = n_F/\alpha_F^2 \).

III. OPTIMAL SCHEDULING FRAMEWORK

Let us assume that there are \( N \) video descriptions to be sent over \( N \) paths from the source to the destination. The most widely used MD coding method in practice is double-description coding for double-path routing, i.e., \( N = 2 \). In general, using more descriptions and paths will increase the robustness against packet losses and path failures, and reduce network congestion due to better load balancing. However, these advantages come at the cost of higher coding redundancy, higher computation complexity, and higher control traffic overhead in the network. As a result, a baseline system having double-description coding with double-path routing will provide significant performance gains at a modest cost [2].

Therefore, we simply split the original sequence into even-indexed and odd-indexed frames and then encode them into two descriptions, \( V_h (h = 1, 2) \). Each frame is compressed into an RTP packet. Let \( R_h \) and \( S_h \) be the code and frame

**B. R-D Hint Information**

Let \( k \) be the index of a packet from a video description, \( V_h (h = 1, 2, ..., N) \). The R-D hint information associated with packet \( k \) consists of the size of packet \( k \) in bits, \( R_h(k) \), and the total increase in MSE distortion, \( D_h(k) \), caused by losing packet \( k \). Assume that packet \( k \) belongs to a Group of Pictures (GOP) defined by the period of the I frames in the compression. Let \( G(k) \) denotes the packet index set corresponding to this GOP. Then, we can compute \( D_h(k) \) as
\[
D_h(k) = \sum_{j \in G(k)} \Delta d_j
\]
(1)
where \( \Delta d_j \) is the increase in MSE distortion associated with packet \( j \) given packet \( k \) is lost. Note that \( \Delta d_j = 0 \) for all \( j \not\in G(k) \).

The computation of \( D_h(k) \) for the compressed packets at encoding time relieves the burden of optimized streaming servers, which can simply read them rather than estimating them on a real-time basis. The size of the packet, \( R_h(k) \), is usually available in packet headers. Figure 2 illustrates the distortion \( \Delta d_j \) for loss of packet \( k \) \((k = 4)\) of the H.264-coded QCIF FORMAN sequence of odd-indexed frames with IntraPeriod=15 and no B frames. For clarity of presentation, each compressed frame is encapsulated into an RTP packet. The simple frame copy method is used to conceal the errors. It can be seen that the MSE per frame ramps up at frame \( k \) and diminishes to zero until the next GOP; it is expected that there are no prior losses before the packet \( k \) and the loss effects are stopped by a new I-frame.

**C. Path Characteristics**

Let \( PT_h \) denote the path between the source and the destination for transmitting description \( V_h \), the path characterized

\[ D_h(k) = \sum_{j \in G(k)} \Delta d_j \]

where \( \Delta d_j \) is the increase in MSE distortion associated with packet \( j \) given packet \( k \) is lost. Note that \( \Delta d_j = 0 \) for all \( j \not\in G(k) \).

The computation of \( D_h(k) \) for the compressed packets at encoding time relieves the burden of optimized streaming servers, which can simply read them rather than estimating them on a real-time basis. The size of the packet, \( R_h(k) \), is usually available in packet headers. Figure 2 illustrates the distortion \( \Delta d_j \) for loss of packet \( k \) \((k = 4)\) of the H.264-coded QCIF FORMAN sequence of odd-indexed frames with IntraPeriod=15 and no B frames. For clarity of presentation, each compressed frame is encapsulated into an RTP packet. The simple frame copy method is used to conceal the errors. It can be seen that the MSE per frame ramps up at frame \( k \) and diminishes to zero until the next GOP; it is expected that there are no prior losses before the packet \( k \) and the loss effects are stopped by a new I-frame.

**Fig. 2.** Loss of frame \( k \) induces distortion in later frames.
rates of $V_h$, respectively (e.g. $r_1 = r_2 = 198$ kbps and $s_1 = s_2 = 15$ fps). Assume the sender at the current time instant has a window $W_h$ of packets from $V_h$ that are considered for transmission along path $PT_h$ of bandwidth $b_h$, bit error rate $e$, and delay $t$. Let $|W_h|$ denote the time length of window $W_h$, i.e., $W_h$ contains $|W_h| * s_h$ frames (e.g. $|W_h| = 1s$). The sender needs to decide omitting/dropping a subset of packets $K_h = \{k_1, k_2, ..., k_n\}$ (if any) from $W_h$ prior to transmission such that its transmission rate does not exceed the assigned bandwidth, i.e.,

$$\sum_{k \in W_h/K_h} R_h(k) \leq b_h$$  \hspace{1cm} (5)

where $R_h(k)$ is the length of packet $k$ in bits and “$\sim” denotes the operator “set difference”. Note that $K_h$ may be empty, if the allocated bandwidth $b_h$ is sufficient to transmit all the packets from $W_h$.

The expected MSE distortion that will affect description $V_h$ if $K_h$ is dropped prior to transmission can be computed as

$$\bar{D}_h = \sum_{j \in W_h} E[D(j)]$$

$$= \sum_{j \in W_h} D(j) + \sum_{j \in W_h/K_h} D(j) \left[1 - (1 - e_h)^{R_h(j)} \cdot P\{t_h < \triangle_h\}\right]$$

where $D(j)$ is the R-D hint information of packet $j$, and $P\{\cdot\}$ is the probability of packet delivery before the deadline that is defined by a forward transfer time (FTT) upper bound $\triangle_h$ (e.g $\triangle = 400$ ms). Note that we assume additivity of the distortions associated with the individual lost packets, ignoring any interdependencies between their effects on the distortion, which does not necessarily hold true when the lost packets are not spaced sufficiently far apart with respect to the intra-refresh period, as recognized in [5]. Still, due to its simplicity and yet good accuracy, the additive model has found a number of applications in streaming and modeling of packetized media [8].

We are interested in finding the best transmission schedules for selecting video packets from each video description to transmit in order to maximize the overall video quality on the receiver side. Therefore, the problem of optimal packet scheduling for MD-coded video streaming (OPT-MD) can be formulated mathematically as follows.

Minimize

$$\bar{D} = \sum_{h=1}^{N} \bar{D}_h$$

$$= \sum_{h=1}^{N} \left\{ \sum_{j \in K_h} D(j) + \sum_{j \in W_h/K_h} D(j) \left[1 - (1 - e_h)^{R_h(j)} \cdot P\{t_h < \triangle_h\}\right] \right\}$$

subject to

$$R_h = \frac{\sum_{k \in W_h/K_h} R_h(k)}{|W_h|} \leq b_h \quad for \quad h = 1, 2, ..., N$$  \hspace{1cm} (8)

The constrained optimization problem OPT-MD can be transformed to a non-constrained problem using $N$ Lagrange multipliers as follows:

$$\min_{K_1, K_2, ..., K_N} \left\{ \bar{D} + \sum_{h=1}^{N} \lambda_h \bar{R}_h \right\}$$

$$= \min_{K_1, K_2, ..., K_N} \left\{ \sum_{h=1}^{N} D_h + \lambda_h \bar{R}_h \right\}$$

$$= \sum_{h=K_h}^{N} \min_{K_h} \left\{ D_h + \lambda_h \bar{R}_h \right\}$$

where the optimal vector of dropping patterns $K^* = \{K^*_1, K^*_2, ..., K^*_N\}$ can be computed by solving for the individual dropping patterns $K^*_h$ as

$$K^*_h = \arg \min_{K_h \in W_h} \left\{ D_h + \lambda_h \bar{R}_h \right\}$$

$$= \arg \min_{K_h \in W_h} \left\{ \sum_{j \in W_h/K_h} D(j) + \sum_{j \in W_h/K_h} D(j) \left[1 - (1 - e_h)^{R_h(j)} \cdot P\{t_h < \triangle_h\}\right] + \lambda_h \cdot R_h(j) \right\}$$

(10)

The solution to (10) is attained by dropping every packet $j \in W_h$, where

$$D(j) < D(j) \left[1 - (1 - e_h)^{R_h(j)} \cdot P\{t_h < \triangle_h\}\right] + \lambda_h \cdot R_h(j)$$

$$\Leftrightarrow \frac{D(j) \left[1 - (1 - e_h)^{R_h(j)} \cdot P\{t_h < \triangle_h\}\right]}{R_h(j)} < \lambda_h$$

(11)

since any packet satisfying (11) must be included into $K^*_h$, otherwise $K^*_h$ is not the optimal solution.

We define the packet utility rate for packet $j$ of description $V_h$ as

$$U_h(j) = \frac{D(j) \left[1 - (1 - e_h)^{R_h(j)} \cdot P\{t_h < \triangle_h\}\right]}{R_h(j)}$$

(12)

From (11), we know that the optimal solution $K^*_h$ for $\lambda^*_h$ contains the optimal solution $K^*_h$ for $\lambda^*_h$ if $\lambda^*_h \geq \lambda_h$.

Then, we have the optimal scheduling algorithm presented in Algorithm 1, where each description decides on which of its own packets should be transmitted such that the end-to-end distortion over all the descriptions is minimized.

IV. EXPERIMENTAL RESULTS

In this section, we compare the performance of our OPT-MD scheduling approach against the conventional packet scheduling strategy, denoted as RANDOM-MD, which randomly drops P-frame packets when making transmission decisions. Performance is measured in terms of the average
luminance peak signal-to-noise ratio (Y-PSNR) in dB of the decoded video frames at the receiver as a function of the path parameters, namely, bandwidth and bit error rate.

Two standard QCIF video test sequences, FORMAN and TABLE, both having 300 frames at 30 fps, are employed in the experiments. Each of them is encoded into two descriptions using JM10.2 of the H.264 video compression standard by splitting its frame sequence into even-indexed and odd-indexed frames. Each compressed frame is encapsulated into an RTP packet. Frame copy is used for error concealment. In all the experiments, packets are considered for transmission in non-overlapping windows of size \( L_w = 30 \) (i.e., \(|W| = 2s\) in (8)). That is, at every transmission instance the sender considers 30 packets from each description for transmission. We simulate bit errors during transmitting packets along a wireless path by randomly flipping certain bits of these packets in terms of the bit error rate of that path. These bit errors are statistically independent. The H.264 decoder simply discards the corrupted packets at the receiving end. We assume that video descriptions are transmitted along two paths, which have the same channel characteristics throughout the following experiments. The decoding deadline is set to 400 ms (i.e., \( \Delta_h = 400\text{ms} \) in (8)).

We examine how OPT-MD and RANDOM-MD adapt to bandwidth variations, when the available bandwidths of the two paths decrease with fixed bit error rates and delay distribution corresponding to approximately 1% overdue packet rate. As seen from Figures 3 and 4, the performance gains reach up to 7.8 dB for FORMAN and 6.8 dB for TABLE for bandwidth of 80–90% of the code rate at the bit error level of 0, 4.9 dB for FORMAN and 6.5 dB for TABLE for bandwidth of 80–90% of the code rate at the bit error level of 10\(^{-7}\), and 4.6 dB for FORMAN and 4.8 dB for TABLE for bandwidth of 80–90% of the code rate at the bit error level of 10\(^{-6}\).

We also examine how OPT-MD and RANDOM-MD adapt to bit errors, when the bit error rates of two paths increase with fixed bandwidths and delay distribution corresponding to approximately 1% overdue packet rate. As seen from Figures 5 and 6, the performance gains reach up to 3.5 dB for FORMAN and 3.5 dB for TABLE for bandwidth of 95% of the code rate at the bit error level of 10\(^{-7}\), 5.5 dB for FORMAN and 6.2 dB for TABLE for bandwidth bandwidth of 85% of code rate at the bit error level of 10\(^{-7}\), and 3.7 dB for FORMAN and 3.5 dB for TABLE for bandwidth bandwidth of 75% code rate at the bit error level of 10\(^{-7}\).

We simulate a time-varying wireless network with dynamic path characteristics, as shown in Figure 7. The detailed information of the path parameters is listed in Table I. Table II gives a comparison of the performance of OPT-MD with that of RANDOM-MD for this time-varying wireless network. We can observe that OPT-MD outperforms RANDOM-MD on an overall basis by 5 dB for FORMAN and 4 dB for TABLE. This demonstrates that by adaptively scheduling packets of the two video descriptions, our method can significantly improve the video quality.

![Fig. 7. The dynamic path characteristics. The second channel status lasts for 4 seconds, while the others last for 2 seconds. The detailed path parameters are listed in Table I.](image)

<p>| TABLE I | DYNAMIC PATH PARAMETERS FOR A TIME-VARYING WIRELESS NETWORK |
|----------------|-------------------------|-------------------------|-------------------------|-------------------------|</p>
<table>
<thead>
<tr>
<th>Status</th>
<th>Bandwidth</th>
<th>Bit Error Rate</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>95%</td>
<td>(10^{-8})</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>85%</td>
<td>(10^{-7})</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>75%</td>
<td>(10^{-6})</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>95%</td>
<td>(10^{-6})</td>
<td>25</td>
</tr>
</tbody>
</table>

<p>| TABLE II | PERFORMANCE COMPARISON BETWEEN OPT-MD AND RANDOM-MD FOR TWO VIDEO SEQUENCES IN A TIME-VARYING NETWORK. |
|----------------|-------------------------|-------------------------|-------------------------|-------------------------|</p>
<table>
<thead>
<tr>
<th>Method</th>
<th>Status 1</th>
<th>Status 2</th>
<th>Status 3</th>
<th>Status 4</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>FORMAN</td>
<td>34.19</td>
<td>28.35</td>
<td>22.38</td>
<td>29.24</td>
<td>28.52</td>
</tr>
<tr>
<td>RANDOM-MD</td>
<td>27.15</td>
<td>21.59</td>
<td>19.78</td>
<td>24.58</td>
<td>23.27</td>
</tr>
<tr>
<td>TABLE</td>
<td>34.93</td>
<td>29.38</td>
<td>23.83</td>
<td>29.46</td>
<td>29.40</td>
</tr>
<tr>
<td>RANDOM-MD</td>
<td>30.12</td>
<td>25.06</td>
<td>20.01</td>
<td>24.89</td>
<td>25.24</td>
</tr>
</tbody>
</table>

V. Conclusion

In this paper, we have studied the packet scheduling problem for multiple-description multiple-path video transport over wireless networks. A rate-distortion optimized packet scheduling algorithm has been proposed to minimize the end-to-end video distortion. The proposed algorithm adaptively selects packets for transmission by simultaneously considering the

---

**Algorithm 1 Optimal scheduling algorithm.**

1. \( K^- = \emptyset \);
2. Compute \( U_h(j) \) for each packet in \( W_h \);
3. Denote the sorted set by \( W^*_h = \{k_1, k_2, \ldots\} \);
4. \( j = 1 \);
5. if \( \sum_{k \in W^*_h} R_h(k) \leq b_h \) then
6. \( \text{return } K^+_h; \)
7. else
8. Add packet \( k_j \in W^t \) to \( K^+_h; \)
9. \( j = j + 1 \);
10. GOTO 5
11. end if
relative importance of a packet and the transmission channel conditions including bandwidth, bit error rate, and delay. Our effort in this work provides an important methodology for realizing high-quality real-time video streaming applications over wireless networks.

REFERENCES


