IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 25, 2019, accepted February 12, 2019, date of publication February 26, 2019, date of current version March 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2900387

Buffer-Driven Rate Control and Packet Distribution for Real-Time Videos in Heterogeneous Wireless Networks

FENG CHEN[1], (Member, IEEE), JIE ZHANG[1], ZHIFENG CHEN[1], (Senior Member, IEEE), JIYAN WU[1], (Member, IEEE), AND NAM LING[1,2], (Fellow, IEEE)

[1]College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, China
[2]Department of Computer Science and Engineering, Santa Clara University, Santa Clara 95053, CA, USA

Corresponding author: Zhifeng Chen (zhifeng@fzu.edu.cn)

This work was supported in part by the National Natural Science Foundation for Young Scholars of China under Grant 61801120, in part by the National Natural Science Foundation of China under Grant 61671153, in part by the National Natural Science Foundation of China under Grant 61561014, and in part by the Fujian Natural Science Foundation under Grant 2018J05104.

ABSTRACT Video transmission bandwidth can be substantially increased by concurrent transmission through multiple radio interfaces introduced by heterogeneous wireless networks. However, delivering high-quality real-time videos with stringent end-to-end delay constraints is still challenging due to the time-varying multipath channel status. Thus, to integrate available multipath bandwidth and guarantee low end-to-end delay, this paper investigates a real-time concurrent multipath video transmission scheme in heterogeneous access networks. First, we present a multipath buffer analysis model (MBAM) to derive the lower and upper bounds of the target video frame bits. Second, given the low-latency and received video quality smoothness, we propose a buffer-driven rate control and packet distribution (BRCPD) algorithm aided by frame-level rate control and multipath transmission buffer awareness based on MBAM. Finally, we implement the proposed application-layer solution BRCPD algorithm in a real-world testbed. The experiment results verify that different from existing multipath video transmission schemes, the proposed algorithm is able to leverage the MBAM-based frame-level rate control to guarantee the target low end-to-end delay. In particular, the BRCPD algorithm is able to guarantee an overdue loss rate of less than 5% under strictly limited delay constraints (250 ms) in the real-world testbed.

INDEX TERMS Heterogeneous wireless networks, concurrent multipath transmission (CMT), rate control, packet distribution, buffer awareness, low-latency.

I. INTRODUCTION

In recent years, the demand for low-latency and high-quality real-time video applications has been increasing rapidly. In the near future, multi-view videos [1], ultra high-definition videos [2], 3D videos [3] and virtual reality [4] video applications are expected to be increasingly prevalent in Internet multimedia applications. These applications require the transmission of real-time videos over wireless networks that can exhibit wide variability in delay, throughput, and coverage. In particular, low latency is a key performance metric in real-time video communication systems, e.g., online games, real-time video chatting, and live sports broadcasting.

However, the current limited single wireless communication capacity is unable to completely satisfy the ever-growing transmission requirements, e.g., current wireless local area networks (WLAN) can provide high throughput but with limited coverage, and cellular networks (Global System for Mobile communications, High Speed Downlink Packet Access, Long Term Evolution) can offer coverage and mobility support but with limited throughput [5]. An increasing number of heterogeneous networks are expected to coexist in future 5G systems to provide superior services [6]. As shown in Figure 1, an effective solution is to consider integrating heterogeneous wireless networks (e.g., WLAN, GSM, HSDPA, LTE) by transmitting real-time videos in those wireless networks simultaneously; such process can be referred to as concurrent multipath transmission (CMT) [7]. CMT can transport data in a parallel

The associate editor coordinating the review of this manuscript and approving it for publication was Hayder Al-Hraishawi.

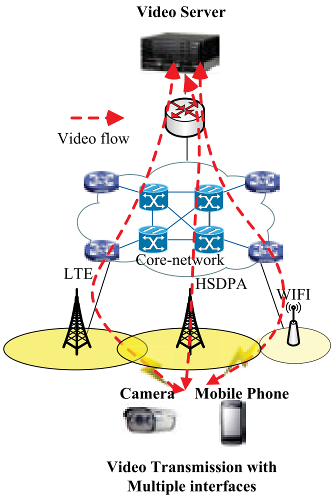

FIGURE 1. Concurrent real-time video transmission in heterogeneous wireless networks.

manner by utilizing multiple network interfaces, and it has been recognized as a promising solution to load balancing and bandwidth aggregation in heterogeneous networks [8].

CMT in heterogeneous wireless networks for real-time videos has become an important research issue over the past few years. Different paths may have heterogeneous performance parameters, such as bandwidth, transmission delay, and packet loss rate. In recent years, research has focused on how to optimize packet scheduling approaches that split video frames into different paths to reduce end-to-end delay and distortion [9]–[14].

A. PROBLEM STATEMENT

Although CMT in heterogeneous wireless networks has been extensively investigated in recent years and video transmission quality can be effectively improved by optimized packet scheduling approaches, new challenges arise in real-time video transmission. As a result of the time-varying status of heterogeneous wireless networks, the performance of CMT may change dynamically, and constant bitrate (CBR) video streaming may not match the bandwidth and lead to low network utilization or expired arrivals. These issues may significantly degrade the quality of real-time videos received. Optimizing the rate control of real-time videos with the purpose of providing low-latency and high-quality videos is a new challenge in CMT. However, to the best of our knowledge, no research work has focused on the optimization of rate control for CMT of real-time videos. Moreover, rate control for CMT is challenged with more difficulties than traditional rate control methods do. These difficulties include the following.

- In the traditional single-path transmission, the video bitrate usually depends on bandwidth directly. However, in CMT, a video frame is split into multiple paths by a packet distribution scheme so that the packets arrive at the receiver simultaneously. Therefore, rate control for CMT needs to be re-analyzed with consideration of the multipath features.
- Frame-level rate control is often adopted in low-latency video communication systems. However, there is no previous study has centered on how to allocate accurate frame bits in CMT. Optimizing the frame bitrate in CMT is a challenge under dynamic bandwidth, delay, and packet distribution schemes.

B. CONTRIBUTIONS

To effectively control the bitrate of real-time video frames in CMT, this paper presents a buffer-driven rate control and packet distributing framework. First, a multipath buffer analysis model (MBAM) is proposed to obtain the upper and lower bounds of a video frame's bitrate under the required end-to-end delay. Second, we formulate the rate control problem by considering the impact of packet distribution and propose a buffer-driven rate control and packet distributing algorithm (BRCPD). Third, we test the BRCPD algorithm on a real-world testbed involving H.264 video streaming. The proposed BRCPD is an application-layer solution and can be implemented without modifications on underlying transmission control protocol/internet protocol (TCP/IP) protocols (e.g., at transport or network layer).

In particular, the contributions of this research can be summarized as follows.

- Different from existing work that focuses on packet distribution frameworks only, the current study proposes the BRCPD framework for multipath video transmission in heterogeneous wireless networks to address the conflict between strict target end-to-end delay and time-varying channel status.
- The MBAM is presented to obtain the lower and upper bounds of target video frame bits in CMT.
- The BRCPD algorithm is designed by choosing the quantization step size in source coding and distributing frame packets to multiple paths on the basis of the MBAM. The proposed algorithm is then tested on a practical platform involving H.264 video streaming. The evaluation results demonstrate that the BRCPD algorithm can guarantee an overdue loss rate of less than 5% under strict delay constraints (250 ms) in the real-world testbed.

C. PAPER ORGANIZATION

The rest of this paper is structured as follows. In Section II, we briefly review and discuss the related work. In Section III, we present the system model and problem formulation. In Section IV, the detailed BRCPD algorithm is described. In Section V, we provide the performance evaluation. In Section VI, we summarize the concluding remarks.

II. RELATED WORK

The recent research efforts relevent to this study can be classified into three categories: 1) concurrent multipath video transmission, 2) rate control for real-time videos,and 3) buffer-driven rate control for video codec.

A. CONCURRENT MULTIPATH VIDEO TRANSMISSION

An increasing number of researchers have concentrated their efforts on flow distribution for concurrent multipath video transmission. Flow distribution can be classified into bit-level distribution and frame-level distribution. In terms of bit-level distribution, Song and Zhuang [10] introduce a Z transform method to model video bitstreams, and analyze the performance of CMT in terms of average delay, delay jitter, and delay outage probability on the basis of probability statistics theory. Xing *et al.* [11] propose a real-time adaptive prediction model for video streaming over multiple wireless access networks. Video bitstreams are modeled with Markov decision process theory and split into multiple paths dynamically. References [10] and [11] show that video quality of service (QoS) can be improved by CMT relative to traditional methods. However, bit-level video packet distribution does not involve the specific video frames in video sequences. Moreover, the end-to-end delay or distortion of each video frame cannot be well guaranteed. Therefore, our earlier works centered on the packet allocation for each video frame with the purpose of improving the end-to-end delay and distortion of video frames [12]–[15]. For example, we establish a video distortion model for each video frame in CMT and propose a frame-level packet distribution algorithm [12]. To guarantee the end-to-end delay for each frame, we build an end-to-end delay model for CMT and a delay-constrained concurrent multipath scheme for high-definition videos to minimize the distortion under the required end-to-end delay [13].

Transmission control at the transport layer is a key research point, which has attracted considerable research attention. Cao *et al.* [16] propose a novel TCP-friendly stream control transmission protocol (SCTP)-based CMT solution with the purpose of achieving fairness in TCP flow and load sharing. Xu *et al.* [17] present an innovative window-based mechanism for flow control to balance delivery fairness and efficiency on the basis of the data-link layer and rate/BW estimation at the transport layer. Han *et al.* [18] consider users' quality of experience (QoE) for mobile video streaming and propose a strategically scheduled delivery of video chunks on the basis of multipath TCP to satisfy user preferences.

Nevertheless, high bit error rates are still unavoidable in heterogeneous wireless networks and forward error correction (FEC) can still be applied to improve the quality of video streaming. Therefore, Tsai *et al.* [19] propose a CMT control scheme that adaptively adjusts the FEC block length and concurrently sends data interleaved over multiple paths. Chilamkurti *et al.* [20] consider a delay-sensitive multipath FEC mechanism. The authors optimize the transmission rate, FEC block length, and FEC redundancy on each path on the basis of the estimation of available bandwidth and a mathematical analytical model. Our previous work in [14] and [21] design an adaptive FEC encoding and distributing algorithm for concurrent multipath video transmission.

As transmission bandwidth increases in CMT, its energy consumption increases as well. Therefore, some researchers have optimized the packet allocation with consideration of the energy efficiency problem [22]–[26]. For instance, Wu *et al.* [22] analyze the energy efficiency problem in CMT for multi-homed terminals and successfully optimize the allocated transmission energy on each link to maximize the living time. In [25], Gao *et al.* provide an application rate-aware energy-efficient subflow management strategy to make a tradeoff between throughput performance and energy consumption for mobile phones; they also consider the QoE of users. Liu *et al.* [26] investigate the energy-efficient resource allocation problem for heterogeneous wireless networks with multihomed user equipments, and propose a two-phase optimization method with higher energy-efficient performance and lower complexity relative to conventional approaches.

B. RATE CONTROL FOR REAL-TIME VIDEOS

Rate control is an important issue in video transmission, especially in time-varying wireless networks. Moreover, the prediction of bandwidth, the determination of sending rate, and congestion control have always been an active and challenging research area in recent years given the requirements of low delay and limited buffer constraints. In general, rate control schemes can be divided into the group of picture level (GOP-level) scheme, frame-level scheme, or macroblock level (MB-level) scheme. The GOP-level scheme allocates bits to each GOP on the basis of the bandwidth, delay, or buffer constraints; In the frame-level scheme, each frame video bits is controlled by an accurate quantization parameter (QP) value that determines, for example, the size of each I frame and P frame. For a given frame, the MB-level scheme controls each $MB's$ bit allocation with target video frame bits.

Rate control algorithms of different levels have been proposed [27]–[31], and some of them are specifically designed for real-time low-delay video transmission, e.g., [29], [31]. In [27], an efficient bit allocation scheme that considers encoder buffer status and frame complexity in low-delay environments is proposed. Specifically, the scheme uses a peak signal-to-noise ratio (PSNR)-based frame complexity estimation and the dynamical Lagrange decisions of the MB-level scheme. Lu *et al.* [28] develop an efficient adaptive bit allocation algorithm based on the GOP-level scheme using reverse dynamic programming (RDP) in which the initial delay value and the corresponding optimal GOP-level bit allocation scheme can be obtained for video streaming. A delay-constrained rate control algorithm is proposed in [29] to achieve a distributed bandwidth sharing with low latency, efficient utilization, and distortion fairness; the video rate of all flows is adapted according to the distortion weight and shadow price controlled by queueing delay.

To maximize the total utility in wireless networks with multiple real-time flows, Zuo *et al.* [30] first divide the goal into submodular optimization problems with many constraints. Moreover, joint rate control and scheduling

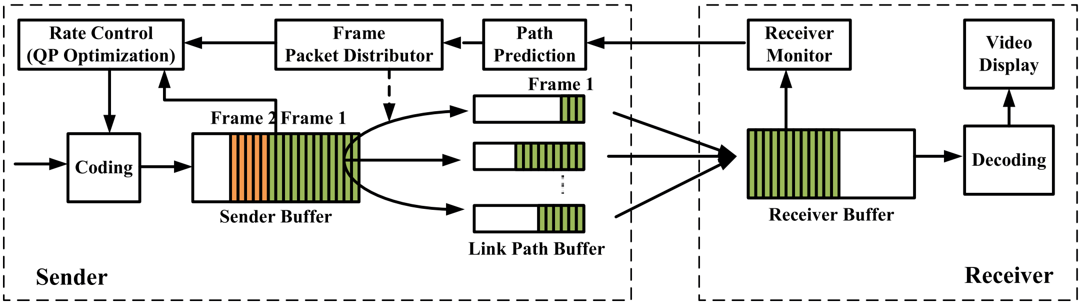

FIGURE 2. Framework of the proposed buffer-driven rate control and packet distribution solution.

policies are designed by each flow dynamically adjusting its traffic load on the basis of observed network status. Eymen *et al.* [31] propose a real-time bandwidth prediction and rate adaption for video calls over cellular networks and modify the H.264 video encoder to produce a hierarchical-P stream with the purpose of controlling encoded videos according to the predicted bandwidth.

C. BUFFER-DRIVEN RATE CONTROL FOR VIDEO CODEC

In most video coding systems, video codec buffers are used to smoothen output video streaming and thereby avoid the delay jitters. A hypothetical reference decoder (HRD) [32] or video buffering verifier (VBV) [33] is often applied to determine the target encoding bits in rate control algorithms, i.e., the target encoding bitstream satisfies no underflow and overflow of sender and receiver buffers. Lee *et al.* [34] propose a CBR encoding method under a limited buffer size to reduce picture quality fluctuations. Considering the variable bitrate (VBR), in our earlier work [35], we analyze a video codec buffer model to obtain the target bitstream for a time-varying channel without buffer underflow and overflow. Shuai and Herfet [36] propose a bitrate adaptation algorithm that considers the buffer delay to support low-latency video streaming. Zhao *et al.* [37] propose a scheme to conduct video bitrate selection on the basis of the current queue buffer status and real-time throughput. A novel QoE-aware video transmission optimization algorithm is proposed in [38] by jointly controlling the transmission rate and playback buffer management to reduce the probability of video playback interruptions and adapt to a constantly changing network status effectively. Kim and Won [39] suggest a frame rate control buffer management technique with respect to buffering delay. Buffer underflow and overflow can be effectively prevented by their technique.

In conclusion, the problem of rate control in CMT of real-time videos has not been addressed in existing research. To the best of our knowledge, the proposed MBAM is the first model to obtain the lower and upper bounds of a target video frame′*s* size under the required network utilization and end-to-end delay in a CMT scenario. Furthermore, the proposed BRCPD is the first algorithm that optimizes rate control for real-time videos in heterogeneous wireless networks with consideration of video quality smoothness and delay jitters.

III. SYSTEM MODEL

A. SYSTEM OVERVIEW

As mentioned previously, video transmission can be improved by CMT. However, due to time-varying wireless channels, core network bandwidth, and round trip time (RTT), packets must be split into multiple paths dynamically. This requirement causes increased variability in video transmission in heterogeneous wireless networks. Therefore, joint rate control and packet distribution are challenging issues in our system. To address the problem, we develop a system framework, which is in Figure 2.

The system framework includes the working components at the sender and receiver sides. At the sender side, the captured real-time video is encoded with H.264/AVC codec (e.g., JM, x264, FFmpeg) in the Coding module, and the coding rate is controlled by the Rate Control module through the optimization of the QP parameter. The encoded video frames are first stored in the Sender Buffer, and the packets of video frames are later distributed to multiple links by the Frame Packet Distributor. In the Frame Packet Distributor module, each video frame data is allocated to different Link Path Buffer with consideration of the performance of each path. In addition, the Path Prediction module in the sender side can predict the bandwidth and RTT for each path based on the basis of the feedback received from the Receiver Monitor at the receiver side.

At the receiver side, video packets can be received through different paths, and out-of-order packets are regrouped into the Receiver Buffer. The received video frame is taken out from the Receiver Buffer at a regular interval and decoded with H.264/AVC codec in the Decoding module. Note that the frames with overdue packets are dropped and replaced with the latest appropriate frame. The Receiver Monitor is responsible for periodically sending back channel status information (i.e., available bandwidth and RTT). In the system, we send the feedback of the Receiver Monitor at each received frame moments.

In our system framework, it can be observed that the target video rate is coupled with the performance of frame

TABLE 1. Basic notations.

Symbol	Definition
N	the number of transmission path.
μ_n	the available bandwidth of path n.
RTT_n	the round trip time of path n.
$\Re$	a set in which underflow and overflow do not occur in the sender and receiver buffers.
$S_e(t)$	the cumulative numbers of encoding bits over time t.
$S_t(t)$	the cumulative numbers of transmitting bits over time t.
$S_r(t)$	the cumulative numbers of receiving bits over time t.
$S_d(t)$	the cumulative numbers of decoding bits over time t.
$d\left(L^k\right)$	the transmitting time needed for the k-th video frame.
B_e	the size of encoder buffer.
B_d	the size of decoder buffer.
$F_e(t)$	the encoder buffer fullness.
$F_d(t)$	the decoder buffer fullness.
t_e^k	the time point adding the k-th video frame bits into encoder buffer.
t_r^k	the time of k-th video frame$'s$ first bit arriving at receiver.
L^k	the k-th video frame bits.
$L_{r,t}^i$	the remaining (unsent) bits of the i-th video frame.
$L_{r,t}^{i,n}$	path-n's allocated bits of the i-th video frame$'$s remaining (unsent) bits.
Ω	the packet allocation scheme.
Δ	the end-to-end delay constraint.
Δt	the time interval of adjacent frames.
L_e	the calculated frame size according to water-filling algorithm.
$\bar{\nu}$	the difference between calculated frame size and real frame size.
λ_L	the minimum value of water-line.
λ	the water-line.
δ_e^0	the initial encoding time time at sender.
δ_t^0	the initial transmitting time at sender.
δ_r^0	the first frame receiving time and decoding time at receiver.
δ_d^0	the initial decoding time at receiver.
δ_e^k	the needed waiting time to send the first bit and sending time of k-th video frame.
δ_t^k	the needed sending time of k-th video frame.
δ_d^k	the needed waiting time to decode k-th video frame at receiver.
P_k	the PSNR of k-th video frame.
Q^k	the quantization parameter of k-th video frame.
T_w	the number of iterations in water-filing algorithm.
T_{ql}	the time complexity for estimating L^k with Q^k.
T_{qp}	the complexity for searching Q^k with L^k.

packet distribution over multiple paths. To settle the coupling, we propose the MBAM to ensure that the video frame$'s$ bitrate does not suffer from overflow and underflow in the sender and receiver buffers. Then, a joint rate control and packet distribution algorithm based on the MBAM is proposed to improve the video transmission performance over multiple paths. The basic notations used in this paper are defined in Table 1.

B. PROBLEM FORMULATION

The ultimate goal of this work is to provide optimal QP decision and packet distribution for each video frame to improve the quality of each video frame gradually under the imposed low end-to-end delay. HRD or VBV is often used in video coding standards to ensure that underflow and overflow do not occur in the sender and receiver buffers. Obviously, overflow in the sender and receiver buffers will cause packet loss, underflow in the sender buffer will cause bandwidth waste, and underflow in the receiver buffer will cause delay jitters. In video encoders, video bitrates are usually controlled to satisfy HRD or VBV and thereby avoid bandwidth waste and delay jitters. In [35], we analyze video codec buffer and delay under a time-varying channel and obtain the lower and upper bounds of the target video frame$'s$ bits without buffer overflow and underflow. Our results in [35] show that delay jitters and video quality can be improved efficiently if channel estimation is possible to encoder.

Suppose $\Re$ represents a set in which underflow and overflow do not occur in the sender and receiver buffers during concurrent multipath video transmission. The problem can be formulated as follows:

$$
\begin{aligned}
&\mathcal{OP}1.\{Q^k, L_n^k\} = \arg\max P_k.\\
&\textit{subject to}:\\
&\qquad C1.L^k \in \Re, \quad \forall k;\\
&\qquad C2.Q^k - Q^{k-1} \leq 1, \quad \forall k;\\
&\qquad C3.\sum_{n=1}^{N} L_n^k = L^k, \quad \forall k;\\
&\qquad C4.L_n^k \geq 0, \quad \forall n, k. \qquad (1)
\end{aligned}
$$

where Q^k is the QP of the k-th video frame and L_n^k denotes the bits allocated to path n for the k-th video frame. P_k denotes the PSNR of the k-th video frame, and L^k represents the number of k-th video frame bits, respectively. To improve the quality smoothly, we suppose that the difference in QP between the k-th video frame and the $k-1$-th video frame should not exceed 1. Obviously, when condition 2 is satisfied, condition 1 may not be satisfied, i.e., no solution may be available for $\mathcal{OP}1$. In this case, $C2$ needs to be relaxed and neglected to guarantee constraint $C1$.

IV. PROPOSED BRCPD SCHEDULING FRAMEWORK

This section presents the scheduling algorithm for optimizing problem $\mathcal{OP}1$. The challenging issues can be summarized as follows.

1) Different from our previous work, the proposed system framework transmits video frame bits on multiple wireless paths. The effects of multiple time-varying channels need to be re-analyzed to obtain set $\Re$.
2) Inspired by our previous work, set $\Re$ is affected by the effective transmission rate, which is associated with the packet distribution scheme. Another challenge is how to jointly consider rate control, packet distribution, and then search for an optimal QP.

To address the problem, we propose the MBAM to obtain constraint set $\Re$. Then, the lower and upper bounds of the target frame bits are estimated with consideration of the packet distribution scheme to simplify term 1 in problem $\mathcal{OP}1$. Finally, a low complexity BRCPD algorithm is proposed for joint rate control and packet distribution on the basis of the above steps.

A. PROPOSED MBAM

To facilitate the analysis, we plot the existing frame bits of the sender and receiver buffers to analyze the buffer status at each moment (Figure 3).

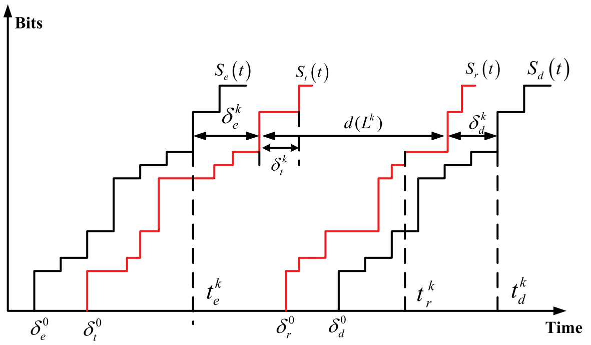

FIGURE 3. Relationship between sender buffer and receiver buffer. $S_e(t)$, $S_t(t)$, $S_r(t)$, and $S_d(t)$ represents the cumulative numbers of encoding, transmitting, receiving, and decoding bits over time t, respectively. δ_e^0 and δ_t^0 denote the initial encoding time and transmitting time at the sender, respectively. δ_r^0 and δ_d^0 denote the receiving time and decoding time of the first frame at the receiver, respectively. t_e^k, t_r^k, and t_d^k indicate the encoding time, first bit receiving time, and decoding time of the k-th video frame, respectively. δ_e^k and δ_t^k denote the waiting time needed to send the first bit and the sending time of the k-th video frame, respectively. δ_d^k is the waiting time needed to decode the k-th video frame at the receiver. $d\left(L^k\right)$ is the transmitting time needed for the k-th video frame that includes the sending time and propagating time.

In H.264/AVC HRD, the decoding process is assumed to be instantaneous. Correspondingly, (Figure 3) we assume that the encoding process is instantaneous. Note that our method is general for non-instantaneous encoding and decoding. We also assume that all bits of one video frame are distributed to multiple paths instantaneously and that the next video frame′s bits can be distributed only after all the bits of this frame have been sent. To briefly analyze the problem, we introduce buffer fullness to illustrate the buffer′s data status. In Figure 3, the buffer fullness of the encoder is defined as $F_e(t) = S_e(t) - S_t(t)$, and that of the decoder is defined as $F_d(t) = S_r(t) - S_d(t)$.

- *Ensure the absence of overflow in the encoder buffer*:

We can easily prove that to avoid overflow in the encoder buffer, the inputted k-th video frame bits shall satisfy

$$L^k \leq B_e - F_e\left(t_e^k\right). \tag{2}$$

The inputted k-th video frame length is constrained by the encoder buffer′s size B_e and fullness at time t_e^k, which is the time point before adding the k-th video frame bits into the encoder buffer.

- *Ensure the absence of underflow in the encoder buffer*:

From Figure 3, we can find that the k-th video frame bits need wait δ_e^k before transmission due to the queue delay of the encoder buffer. The real transmission time is denoted as δ_t^k, which is associated with the CMT bandwidth and the k-th video frame bits. To avoid underflow in the encoder buffer, we should make sure that the k-th video frame bits have not being completely transmitted before the $k+1$-th video frame bits arrive. Then, we can easily prove that the condition satisfying the encoder buffer with no underflow is

$$\delta_t^k + \delta_e^k \geq t_e^{k+1} - t_e^k. \tag{3}$$

Obviously, $t_e^{k+1} - t_e^k$ is constant and related to video frame rate. δ_e^k is the cumulative transmission time of the rest of the frames in the encoder buffer. Let $L_{r,t}^i$ and $\delta_{r,t}^i$ denote the remaining (unsent) bits of the i-th video frame and the corresponding transmission time required, respectively; δ_e^k can be calculated with

$$\delta_e^k = \sum\nolimits_{i=k_{left}}^{k-1} \delta_{r,t}^i, \tag{4}$$

in which k_{left} indicates the current sending frame at the time point before adding the k-th video frame bits into the encoder buffer. Obviously, when $i \neq k_{left}$, $L_{r,t}^i = L^i$. Different from that in single-path transmission, the sending time $\delta_{r,t}^i$ in multipath transmission is

$$\delta_{r,t}^i = \max \frac{L_{r,t}^{i,n}}{\mu_n}, \tag{5}$$

where $L_{r,t}^{i,n}$ is path-n′s allocated bits of the i-th video frame′s remaining (unsent) bits; it is associated with the packet allocation scheme. In other words, δ_e^k can be obtained with a known packet allocation scheme.

Therefore, to guarantee the absence of underflow in the encoder buffer, the required sending time of the k-th video frame δ_t^k should satisfy

$$\delta_t^k \geq t_e^{k+1} - t_e^k - \sum\nolimits_{i=k_{left}}^{k-1} \max \frac{L_{r,t}^{i,n}}{\mu_n}. \tag{6}$$

From (4) and (5), we can find that the transmission time of the k-th video frame bits monotonously changes with its bit length under a unified packet allocation scheme. Therefore, we can easily search the minimum L^k by trying different values iteratively under a unified packet allocation scheme. Suppose Ω_1 is defined as the searching method. We can then indicate the encoder buffer condition without underflow as

$$L^k \geq \Omega_1\left(t_e^{k+1} - t_e^k - \sum\nolimits_{i=k_{left}}^{k-1} \max \frac{L_{r,t}^{i,n}}{\mu_n}\right). \tag{7}$$

- *Ensure the absence of underflow in the decoder buffer*:

To avoid underflow in the decoder buffer, we need to make sure that all of the $k+1$-th video frame bits have arrived at the receiver before the k-th video frame′s decoding process starts. Furthermore, the end-to-end delay from the encoding process at the sender to the decoding process at the receiver is constant once the system runs. In this work, such delay is denoted as Δ and is associated with δ_e^0 and δ_d^0, i.e., $\Delta = \delta_d^0 - \delta_e^0$. Therefore, δ_e^0 and δ_d^0 exert critical impact on the end-to-end delay performance. In H.264/AVC, δ_e^0 is called *initial cpb removal delay*, i.e., the initial encoding time, and δ_d^0 is called *initial cpb removal delay* offset, i.e., the initial transmission time at the sender. Both parameters are defined in supplemental enhancement information on the buffering period [32].

As a result, we find that the important work is to estimate the time needed for all of the k-th video frame bits arriving at the receiver and to guarantee that the needed time does not exceed the target end-to-end delay Δ. From Figure 3, it is easy to prove that the needed time is equal to the sum of the waiting time in the sender buffer and the transmission time from the sender to the receiver. Then, we can have

$$\delta_e^k + d\left(L^k\right) \leq \Delta, \tag{8}$$

where $d\left(L^k\right)$ represents the transmission time for all of the k-th video frame bits arriving at the receiver. Obviously, the multipath transmission time of the k-th video frame should be chosen as the maximum transmission time from the video packet allocated paths. Using the queue model, we can write the function as

$$d\left(L^k\right) = \max\left(\frac{L_n^k}{\mu_n} + RTT_n/2\right), \tag{9}$$

where $RTT_n/2$ is the one-way delay of the n-th path. L_n^k is the path-n's allocated bits of the k-th video frame and is associated with the packet allocation scheme. Then, from (4) and (9), we can rewritten (8) as

$$\max\left(\frac{L_n^k}{\mu_n} + RTT_n/2\right) \leq \Delta - \sum\nolimits_{i=k_{left}}^{k-1} \max \frac{L_{r,t}^{i,n}}{\mu_n}. \tag{10}$$

Similarly, according to the monotonic nature identified and using the iterative searching method, we can conclude that to ensure the absence of underflow in the decoder buffer, the input k-th video frame bits shall satisfy

$$L^k \leq \Omega_2\left(\Delta - \sum\nolimits_{i=k_{left}}^{k-1} \max \frac{L_{r,t}^{i,n}}{\mu_n}, RTT_n\right), \tag{11}$$

where Ω_2 is defined as the searching method.

- *Ensure the absence of overflow in the decoder buffer*:

To simplify the problem, we ensure that the decoder buffer has no overflow by satisfying the condition in which the space is enough for all the k-th video frame bits when the first bit arrives at the receiver. In other words, the left space of the receiver buffer shall be no less than the length of the k-th video frame bits when the first bit arrives. From Figure 3, we can have

$$F_d\left(t_r^k\right) + L^k \leq B_d, \tag{12}$$

where t_r^k denotes the time when the k-th video frame$'s$ first bit arrives at the receiver and B_d is the size of the receiver buffer. In multipath transmission, the first bit to arrive shall be transmitted through the path with minimum delay; therefore, t_r^k shall be calculated with

$$t_r^k = t_e^k + \delta_e^k + \min_n\left(RTT_n/2\right). \tag{13}$$

As mentioned previously, $F\left(t_r^k\right)$ can be obtained from

$$F\left(t_r^k\right) = S_r\left(t_r^k\right) - S_d\left(t_r^k\right). \tag{14}$$

Obviously, the calculated number of bits arriving at the receiver before t_r^k can be obtained by calculating the total number of bits of the previous $k-1$ frames, i.e., $S_r\left(t_r^k\right) = \sum_{i=0}^{k-1} L^i$. Figure 3 shows that the decoder and encoder schedules become connected as follows:

$$S_d\left(t\right) = S_e\left(t - \Delta\right). \tag{15}$$

Therefore, $S_d\left(t_r^k\right)$ can be obtained with

$$S_d\left(t_r^k\right) = S_e\left(t_r^k - \Delta\right) = \sum\nolimits_{i=0}^{\left(t_r^k - \Delta - \delta_e^0\right)/\Delta t} L^i, \tag{16}$$

where Δt represents the frame interval time. Then, (12) can be rewritten as

$$L^k \leq B_d + \sum\nolimits_{i=0}^{\left(t_r^k - \Delta - \delta_e^0\right)/\Delta t} L^i - \sum\nolimits_{i=0}^{k-1} L^i. \tag{17}$$

B. LOWER AND UPPER BOUND OF L^K

Through the previous analysis, we find that the upper and lower bounds for L^k can be obtained from (2), (7), (11), and (17) to simplify $C1$ in $\mathcal{OP}1$. However, (7) and (11) depend on the packet allocation scheme. Therefore, we need to design a packet allocation scheme and obtain the upper and lower bounds for L^k on the basis of the scheme.

To determine the packet allocation scheme, we can aim to minimize the end-to-end transmission delay for the k-th video frame, i.e., $\min d\left(L^k\right)$. Therefore, we can formulate a sub-problem to search the optimal packet allocation as

$$\begin{aligned}
&\mathcal{OP}2.\{L_n^k\} = \arg\min d\left(L^k\right).\\
&\quad subject\ to: C1.d\left(L^k\right) = \max_n\left(\frac{L_n^k}{\mu_n} + \frac{RTT_n}{2}\right), \quad \forall k;\\
&\qquad\qquad C2.\sum_{n=1}^{N} L_n^k = L^k, \quad \forall k;\\
&\qquad\qquad C3.L_n^k \geq 0, \quad \forall n, k.
\end{aligned} \tag{18}$$

The formulation indicates that problem $\mathcal{OP}2$ can be easily solved by the water-filling algorithm [40].

Algorithm 1 Packet Distribution

```
Input: $\mu_n, \forall n$; $RTT_n, \forall n$; $\bar{\nu}$; $L^k$
Output: $L_n^k$
Initialize $d_{\min} = \min_n (RTT_n/2)$; $\tilde{d} = \max_n (RTT_n/2)$;
            $L_e = 0$; $\nu = \bar{\nu}$.
while $(L_e - L^k \geq \nu \parallel L^k - L_e \geq \nu)$ do
    $L_n^k \Leftarrow \mu_n\left(\tilde{d} - RTT_n/2\right), L_e \Leftarrow \sum_{n=1}^{N} L_n^k$.
    if $L_e - L^k \geq \nu$ then $\tilde{d} \Leftarrow \left(d_{\min} + \tilde{d}\right)/2$.
    else if $L^k - L_e \geq \nu$ then $\tilde{d} \Leftarrow \tilde{d}*2$.
    else return $L_n^k$
end
```

Note that in the water-filling algorithm, the end-to-end delay is the same across all paths, i.e., the water-line is the end-to-end delay of each path. According to the water-line, we can obtain the corresponding L_n^k as

$$L_n^k = \mu_n\left(\lambda - RTT_n/2\right), \tag{19}$$

where λ represents the water-line in the water-filling algorithm. Obviously, the optimal L_n^k can be searched by adjusting λ iteratively when $L^k = \sum_{n=1}^{N} L_n^k$. The detailed searching method can be found in Algorithm 1.

Therefore we can obtain the lower bound by transforming (7) as

$$L_{\min}^k = \sum_{n=1}^{N} \mu_n \left(\lambda_L - RTT_n/2\right), \tag{20}$$

$$\lambda_L = t_e^{k+1} - t_e^k - \sum\nolimits_{i=k_{left}}^{k-1} \max L_{r,t}^{i,n} \Big/ \mu_n. \tag{21}$$

Similarly, we can rewrite (11) as

$$L^k \leq \sum_{n=1}^{N} \mu_n \left(\Delta - \sum\nolimits_{i=k_{left}}^{k-1} \max L_n^i \Big/ \mu_n - RTT_n/2\right). \tag{22}$$

The upper bound $L_{\max}^k$ can be obtained as the minimized value of (2), (17), and (22), i.e., the upper bound can be simplified as

$$\begin{aligned} L_{\max}^k = \min\{B_e - F_e\left(t_e^k\right), B_d + S_e\left(t_r^k - \Delta\right) \\ - S_r\left(t_r^k\right), \sum_{n=1}^{N} \mu_n \left(\Delta - \delta_e^k - RTT_n/2\right)\}, \end{aligned} \tag{23}$$

where the available bandwidth μ_n can be predicted according to the method in [31], which will be discussed later.

C. PROPOSED BRCPD

In this subsection, we propose a low-complexity BRCPD algorithm on the basis of the MBAM and the corresponding lower and upper bounds of L^k. In addition, the QP estimation and path status estimation in the BPCPD algorithm are introduced in this subsection.

- ***BRCPD algorithm***

Given the lower and upper bounds of L^k, problem $\mathcal{OP}1$ can be written as

$$\begin{aligned} \mathcal{OP}3.\{Q^k,\ L_n^k\} &= \arg\max P_k. \\ \textit{subject to}: C1.L_{\min}^k &\leq L^k \leq L_{\max}^k, \quad \forall k; \\ C2.Q^k - Q^{k-1} &\leq 1, \quad \forall k; \\ C3.\sum_{n=1}^{N} L_n^k &= L^k, \quad \forall k; \\ C4.L_n^k &\geq 0, \quad \forall n, k. \end{aligned} \tag{24}$$

The smaller QP should be chosen to obtain the best video quality. Therefore, without considering $C1$ in $\mathcal{OP}3$, the optimal QP, which is named Q_{opt}^k, should be taken with $Q^{k-1} - 1$ owing to the $C2$ constraints in $\mathcal{OP}3$. In other words, if QP $= Q^{k-1} - 1$ and the corresponding $\tilde{L}^k$ is able to meet $C1$, then the optimal Q_{opt}^k can be obtained with $Q^{k-1} - 1$ directly. However, if the corresponding $\tilde{L}^k$ cannot meet $C1$, the optimal QP should be chosen from the constraint $C1$ to guarantee the end-to-end delay, i.e., $C2$ in $\mathcal{OP}3$ should be relaxed. When the corresponding $\tilde{L}^k$ cannot meet $C1$, the target L_{opt}^k value should be chosen as the value closest to $\tilde{L}^k$ to reduce the gap between Q^k and Q^{k-1}, i.e., if $\tilde{L}^k > L_{\max}^k$, then $L_{opt}^k = L_{\max}^k$; if $\tilde{L}^k < L_{\min}^k$, then $L_{opt}^k = L_{\min}^k$.

Therefore, we can propose the solution for $\mathcal{OP}3$ in the flowing proposition.

Proposition 1: For the QP control and packet allocation problem with the MBAM constraint, the following steps can be taken to obtain the optimal solution:

1) Set the initial Q^k with $Q^{k-1} - 1$, and check whether the corresponding $\tilde{L}^k$ is able to meet the MBAM constraint. If it is true, then the corresponding Q^k is the optimal solution.
2) If the corresponding $\tilde{L}^k$ is greater than the upper bound $L_{\max}^k$, select the upper bound $L_{\max}^k$ as the optimal solution. Conversely, if $\tilde{L}^k$ is less than the lower bound $L_{\min}^k$, select the lower bound $L_{\min}^k$ as the optimal solution.

The detailed algorithm for Proposition 1 is presented in Algorithm 2.

Algorithm 2 BRCPD Algorithm

```
**Input:** $\mu_n, \forall n; RTT_n, \forall n; L^{k_{left}}; B_e; B_d; \delta_d^0; \delta_e^0 = 0$
**Output:** $Q_{opt}^k$
**Initialize** $Q^k = Q^{k-1} - 1, \Delta = \delta_d^0 - \delta_e^0, t_e^k = k * \Delta t.$
**Calculate** $\tilde{L}^k$ **with** $Q^k$ **by** (25) **and** (26).
**Procedure:** $\delta_t^k$ **and** $\delta_e^k$ **estimation.**
    **For** $i \Leftarrow k_{left} tok$ **do**
        **Determine** $L_n^i$ **with algorithm 1;**
            $\delta_t^i \Leftarrow \max \frac{L_n^i}{\mu_n}, \delta_e^i \Leftarrow \sum_{i=k_{left}}^{k-1} \delta_t^i$
    **end**
**Calculate** $L_{\max}^k$ **and** $L_{\min}^k$ **by** (20) **and** (23).
**If** $L_{\min}^k \leq \tilde{L}^k \leq L_{\max}^k$ **then** $Q_{opt}^k = Q^k$
    **else if** $\tilde{L}^k > L_{\max}^k$ **then** $L_{opt}^k = L_{\max}^k$
    **else if** $\tilde{L}^k < L_{\max}^k$ **then** $L_{opt}^k = L_{\min}^k$
**end**
**Search** $Q_{opt}^k$ **with** $L_{opt}^k$.
```

To verify the maneuverability of the proposed algorithm, we analyze its complexity in Proposition 2.

Proposition 2: The worst-case time complexity of Algorithm 2 is $o\left(T_{ql} + T_w \cdot N \cdot \left(k - k_{left}\right) + T_{qp}\right)$, *in which* T_w *is the number of iterations involved in video packet allocation using the water-filling algorithm.*

Proof: Suppose the time complexity for estimating L^k with Q^k is T_{ql}, and the complexity for searching Q^k with L^k is T_{qp}. To obtain the upper and lower bounds of L^k, we calculate δ_t^k and δ_e^k by using the water-filling algorithm for each video frame. Suppose the number of iterations in the water-filing algorithm is T_w. Then, the complexity of calculating δ_t^k and δ_e^k is $T_w \cdot N \cdot \left(k - k_{left}\right)$, where N is the number of transmission paths. Consequently, the complexity of Algorithm 2 is $o\left(T_{ql} + T_w \cdot N \cdot \left(k - k_{left}\right) + T_{qp}\right)$.

We can conclude that the complexity of our proposed algorithm is linear relative to the number of paths and the number of remaining frames in the sender buffer, i.e., the proposed algorithm can be well operated in an embedded system due to its low complexity.

- *QP Estimation*

According to Algorithm 2, to realize the rate control of the frame level, we are bound to derive the relationship between frame size and quantization step size. Obviously, the smaller quantization step size is, the larger frame size is. To characterize this relationship, we denote the following equation referring to [41]:

$$L^k = \alpha q^{-2} + \beta q^{-1} + \gamma, \tag{25}$$

in which q is the quantization step size. α and β are coefficients that are obtained by a large number of experiments and least squares linear fitting. Simultaneously, the resulting relationship between the quantization parameter and the equivalent quantization step size is now given by [42]

$$q = \left(2^{\frac{1}{6}}\right)^{Q-4}, \tag{26}$$

where Q is the quantization parameter. The parameter is used to determine the quantization of the transform coefficients in H.264/AVC, and it can be taken with 52 values [43].

- *Path Status Estimation*

To guarantee the performance of the BRCPD algorithm, it is important to accurately estimate each path's status, such as bandwidth and RTT. In our system, similar to the NTP protocol [44], we calculate RTT by marking all packets' sending time and receiving time at the sender and receiver, respectively. Effective bandwidth prediction is difficult in wireless networks. In [13], the pathChirp algorithm [45] is employed to estimate the channel status. However, an obvious problem is that using the pathChirp algorithm to predict bandwidth wastes additional bandwidth resources for sending probe messages. This feature significantly reduces the transmission efficiency at low throughput rates. In the current work, a real-time bandwidth prediction algorithm [31] is adopted by using video frame packets as the probe packets directly. This method avoids the action of sending redundant probe messages. In addition, this method uses an online linear adaptive filter to predict the next time bandwidth according to the detected historical bandwidth.

In our system, the proposed bandwidth prediction algorithm in [31] is tested in a real-world testbed based on the HOLOWAN network simulator [46]. At the sender side, we directly packetize the video frame and then send them together in a burst. The resulting instantaneous sending rate is likely to be higher than the instantaneous capacity of the link is. Consequently, packets queue up in the bottleneck. At the receiver side, we take capacity measurements by calculating the result of the frame size dividing the period just past. Then, the receiver sends a small feedback packet back to the source, and the sender uses the latest feedback received to predict the available bandwidth in the next period. The prediction performance in LTE and WIFI networks is depicted in Figure 4 and Figure 5, respectively. We test our prediction algorithm with an LTE/WIFI interface in the real-world testbed at a moving speed of 20 km/h. Then, the captured bandwidth/RTT is imported into the HOLOWAN network simulator, and the real-world network bandwidth/RTT is reproduced later. Thereafter, we replace the real interfaces with the HOLOWAN network simulator and test our prediction algorithm again. From Figure 4 and Figure 5, we can observe that the bandwidth/RTT of the LTE and WIFI channels can be well predicted.

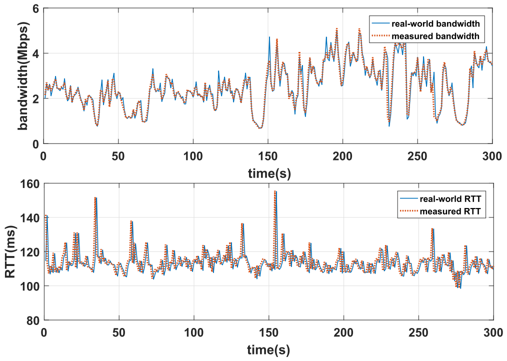

FIGURE 4. Real-world and measured bandwidth (top) / RTT (bottom) over LTE in HOLOWAN network simulator.

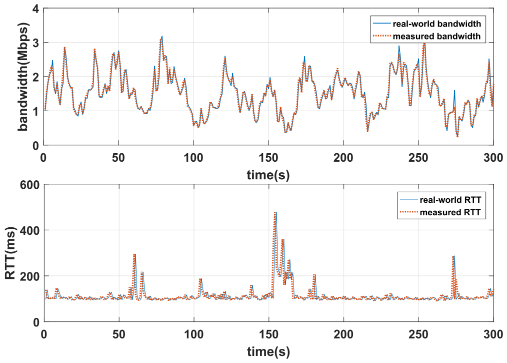

FIGURE 5. Real-world and measured bandwidth (top) / RTT (bottom) over WIFI in the HOLOWAN network simulator.

V. PERFORMANCE EVALUATION

In this section, extensive experiments are conducted to evaluate the performance of the proposed algorithm in a real-world testbed. We first describe the evaluation methodology that includes the real-world testbed setup, reference schemes, and performance metrics. Then, the evaluation results are depicted and discussed in detail.

A. *EVALUATION METHODOLOGY*

1) REAL-WORLD TESTBED SETUP

As shown in Figure 6, a real-world testbed is developed for real-time videos in CMT. The testbed includes a front-end embedded multihomed client (Sender) and a back-end server (Receiver). The sender works under a Linux embedded platform with arm, six-core 2.0 GHz CPU, and 4*GB* RAM. The

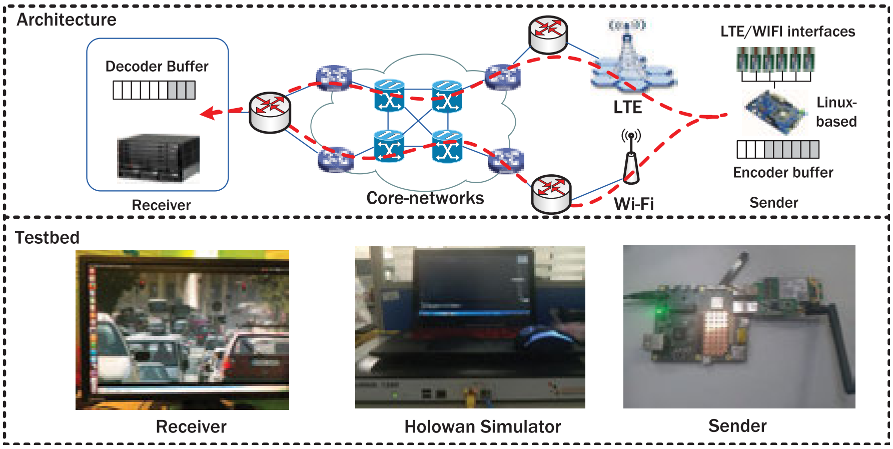

FIGURE 6. Experimental testbed for multipath video transmission.

FIGURE 7. The video test sequences. (a) ParkScene. (b) Rush hour. (c) Bosphorus. (d) Old town cross.

receiver works on a Linux platform with Intel, eight-core 3.6 GHz CPU and 8*GB* RAM. At the sender, a test video is encoded and pushed into the sender buffer, the encoded video can be concurrently sent through LTE and WIFI paths to the receiver, which is connected to the internet with a Gigabit interface. The User Datagram Protocol (UDP) is adopted for data transmission. We first test and record the available bandwidth and RTT of the LTE/WIFI path in a real mobile environment. Then, the network condition is recurred by the HOLOWAN network simulator.

H.264/AVC reference software X.264 is adopted as the video encoder. The uncoded video is compressed at 25 frames per second, and a GoP consists of 25 frames. The GoP structure is $IP \dots P$. Each of the test sequences features a different pattern of temporal motion and spatial characteristics. The video test sequences are *ParkScene, Rush hour, Bosphorus,* and *Old town cross* in 1080P format (Figure 7). To facilitate tagging later, we abbreviate these test sequences as *Park, Rush, Bosp,* and *Town*. In order to obtain statistically meaningful results, we concatenate the test sequences several times to be 20000-frame long.

From Figure 4 and Figure 5, we can observe that the basic RTT in our test system is around 100 ms due to the wide-area network delay. Therefore, in our system, the target end-to-end delay is set to 250 ms, i.e., the target time from the frame finishing the encoding process at the sender to the completion of the receiving process at the receiver for each frame is set to 250 ms.

2) REFERENCE SCHEMES

To verify the superiority of our proposed algorithm in rate control, we compare its performance with that of the constant bit rate scheme and GOP-level bit control scheme.

a: DRA-CBR

A scheduling framework allocates the packets at frame-level for constant rate video with the objective of reducing end-to-end delay and increasing goodput. The constant rate is chosen as the average total bandwidth of paths. The packet allocation scheme refers to [13] partly.

b: DRA-GOP

Different from DRA-CBR, DRA-GOP executes a dynamic rate assignment at GOP-level on the basis of multipath total transmission capacity. In DRA-GOP, the video rate is changed with the measured total bandwidth of all paths per second.

3) PERFORMANCE METRICS

- ***End-to-end Delay***: The end-to-end delay is counted from the frame pushed into the buffer at the sender to all of the packets arriving at the receiver. To accurately measure end-to-end delay, the sender and receiver are synchronized using the NTP protocol by a direct connection through wired Ethernet interfaces.
- ***PSNR***: PSNR is a popular metric of received video quality, and it can be obtained by comparing the original and received video frame. To conceal the lost packets and expired packets, we replace the frame by copying the last received frame.
- ***Goodput***: Goodput is the number of useful bits arriving at the receiver under the target end-to-end delay,

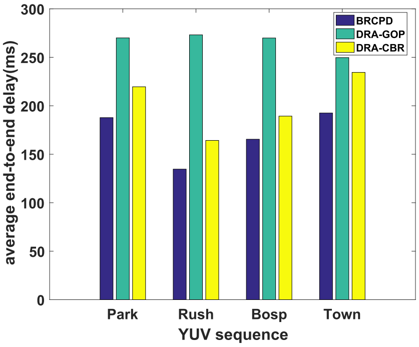

FIGURE 8. Comparison of average end-to-end delays in different video sequences.

i.e., the number of sent bits excludes the lost packets and expired packets. Goodput can also be seen as the application-level throughput [47].

- ***Bandwidth Consumption***: Bandwidth Consumption (BC) is the total number of bits sent on all paths. Different from the total bandwidth, bandwidth consumption represents the actual bandwidth used per second of the total paths.
- ***Overdue Loss Rate***: Overdue loss rate represents the loss probability of packets arriving at the receiver out of the required end-to-end delay.
- ***Buffer Status***: Buffer status is the underflow and overflow probability of the sender buffer and receiver buffer.

B. EVALUATION RESULT

1) END-TO-END DELAY

Figure 8 presents the evolution in terms of average end-to-end delays in different video sequences. It can be seen that most end-to-end delays are guaranteed with 250 ms by our proposed algorithm. The proposed algorithm outperforms the other reference approaches in terms of the end-to-end delay metric. This difference is due to DRA-CBR operating with a constant video rate, which will cause terrible congestion when the bandwidth is insufficient. DRA-GOP working at the GOP-level may generate large I frame bits, which may produce a large end-to-end delay. Moreover, the controlled video rate of the DRA-GOP scheme based on the total bandwidth of all paths is greater than the effective throughput is, obviously. The result shows that the total bandwidth-based GOP-level rate control is inappropriate for multipath video transmission. Figure 9 shows the evolution of instantaneous end-to-end delays of the video frames indexed from 3000 to 6000 in the *Park* sequence. It can be observed that the proposed BRCPD algorithm achieves better performance in low end-to-end delay protection, By contrast, the end-to-end delay of DRA-GOP and DRA-CBR suffers from jitters due to the time-varying channel status.

To clearly understand the distribution results depicted in Figure 9, we plot the cumulative distribution function (CDF) of end-to-end delay in Figure 10 for comparison. We can observe from these statistical results that the proposed BRCPD algorithm guarantees the delivery of nearly 90 percent of the video frames within 250 ms in the *Park* sequence. By contrast, DRA-CBR only achieves about 60%, and DRA-GOP only obtains about 50%.

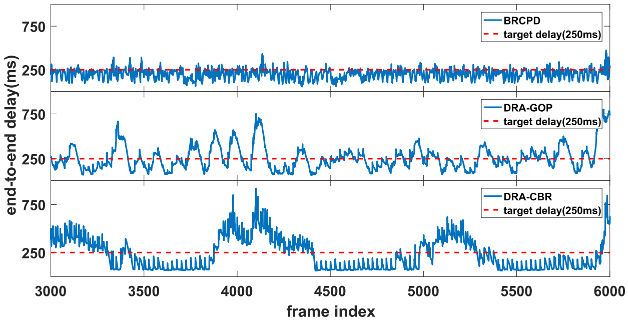

FIGURE 9. Comparison of instantaneous end-to-end delays of video frames indexed from 3000 to 6000 in *Park* sequence.

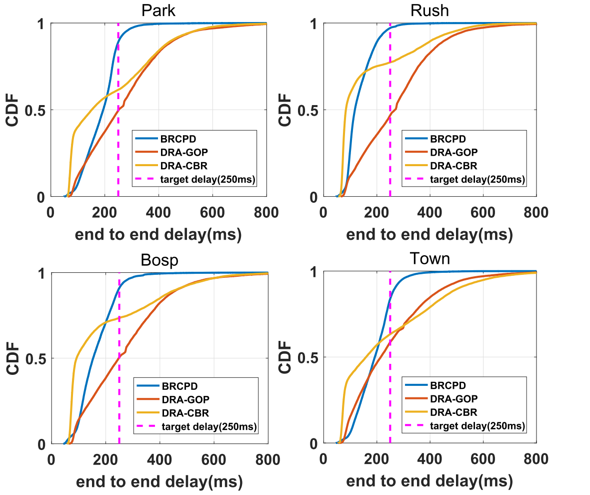

FIGURE 10. CDF of end-to-end delay in different video sequences. The target end-to-end delay is set to 250 ms.

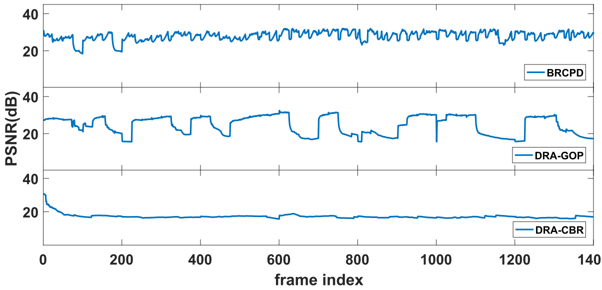

FIGURE 11. Comparison of instantaneous PSNR values of video frames indexed from 0 to 1400 in *Park* sequence. The target end-to-end delay is set to 250 ms.

2) PSNR

Figure 11 demonstrates the instantaneous PSNR performance of the BRCPD algorithm and other reference approaches in the *Park* sequence. We can see that the proposed BRCPD algorithm is able to transmit much smoother video quality relative to DRA-GOP, and its PSNR performance outper-

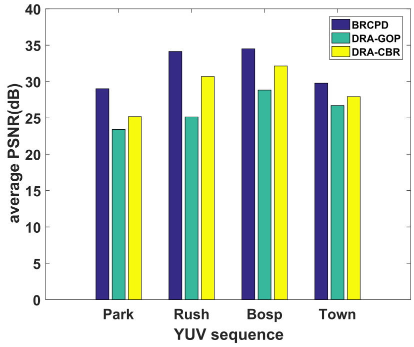

FIGURE 12. Comparison of average PSNR in different video sequences.

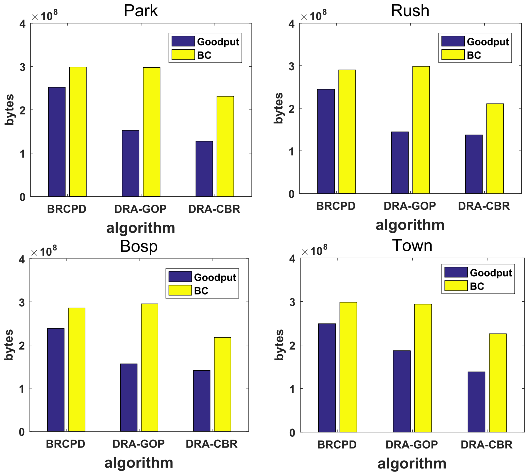

FIGURE 13. Comparison of goodput and bandwidth consumption (BC) performance in different video sequences.

forms DRA-CBR. Figure 12 shows the average PSNR performance in different video sequences. The results show that the proposed BRCPD algorithm always achieves higher values than the other reference approaches do in different video sequences.

3) GOODPUT/BANDWIDTH CONSUMPTION

To present the transmission effectiveness of the proposed algorithm, we plot the goodput and bandwidth consumption comparison for different video sequences in Figure 13. We can find that, in the proposed BRCPD algorithm, the gaps between the goodput and bandwidth consumption are small for different video sequences, whereas the gaps in DRA-CBR and DRA-GOP are larger than that in our proposed algorithm. It shows the superior performance of the proposed algorithm in terms of bandwidth utilization.

4) OVERDUE LOSS RATE

To clearly present the overdue loss rate performance, we plot the average overdue loss rate for different sequences in Figure 14, and the average overdue loss rate for different end-to-end delay constraints in Figure 15 (*Rush* sequence). As shown in Figure 14, the overdue loss rate can be improved markedly by our proposed algorithm under a strict low-latency constraint. Moreover, even when the end-to-end delay constraint is low (e.g., 200 ms), the proposed scheme still performs well, as verified in Figure 15. By contrast, the performance of the reference schemes sharply declines when the end-to-end delay constraint is low.

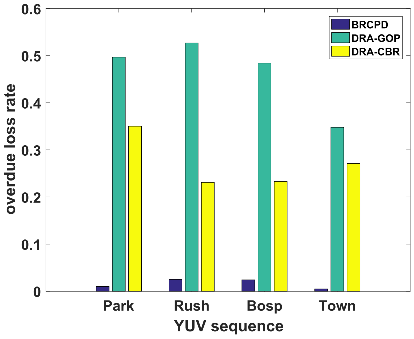

FIGURE 14. Comparison of overdue rate performance in different video sequences.

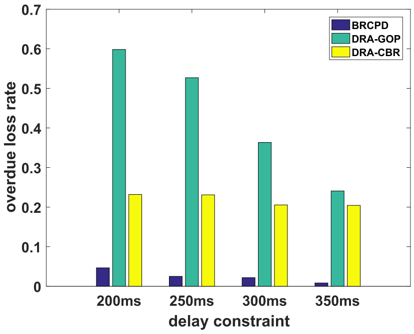

FIGURE 15. Comparison of overdue rate performance for different end-to-end delay constraints.

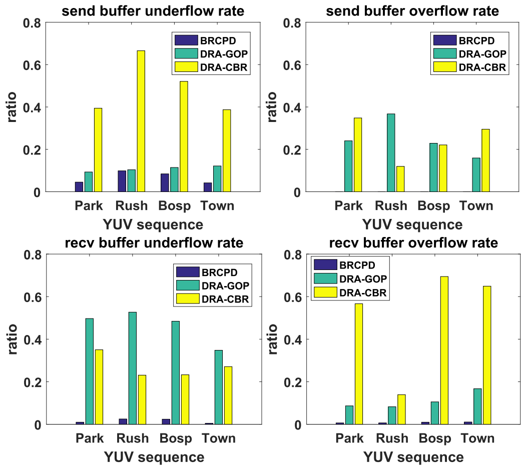

FIGURE 16. Sender buffer and receiver buffer statuses in different video sequences.

5) BUFFER STATUS

In Figure 16, we compare the buffer status, including the underflow rate and overflow rate of the sender buffer and

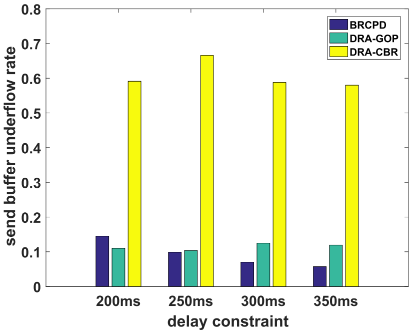

FIGURE 17. Comparison of sender buffer's underflow rate for different required end-to-end delay.

receiver buffer, of the BRCPD algorithm and other reference algorithms for different video sequences. From Figure 16, we can be observe that a substantial improvement in preventing buffer underflow and overflow can be achieved with the proposed BRCPD algorithm. From the overflow status of the sender and receiver in Figure 16, we can find that our proposed algorithm can completely avoid overflow in the sender buffer and receiver buffer. However, in terms of underflow status, underflow still occurs due to the limited end-to-end delay, i.e., low bitrate should be chosen to guarantee low end-to-end delay, which might result in some paths being idle. As shown in Figure 17, the underflow rate of the sender buffer in the *Rush* sequence declines when the required end-to-end delay increases. This result demonstrates the great superiority of the MBAM in the proposed algorithm.

In conclusion, the superiority of the proposed BRCPD algorithm can be summarized as follows. First, low end-to-end delay can be well guaranteed, and PSNR performance is improved. Second, low invalid bandwidth consumption can be obtained because of the low overdue loss rate. Finally, an obvious improvement in buffer status, especially for overflow, can be achieved.

VI. CONCLUSION

In this paper, we propose an improved video rate control issue for real-time video CMT over heterogeneous wireless networks. We derive the lower and upper bounds of video frame bits given the buffer queue and the target end-to-end delay constraint. To improve the smoothness of the received video quality, we further develop the BRCPD algorithm to dynamically adjust video rate at the frame level and allocate packets into multipaths. The real-world testbed demonstrates that the proposed BRCPD algorithm outperforms the existing schemes with minimal end-to-end delay and high received video quality. In our future research, we will consider 1) designing a buffer space reservation for long-term frame bits on the basis of the effective path prediction and 2) jointly optimizing FEC and rate control in our system.

REFERENCES

[1] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou, "Multi-view video summarization," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 717–729, Nov. 2010.

[2] Y. Ye and P. Andrivon, "The scalable extensions of HEVC for ultra-high-definition video delivery," *IEEE Multimedia*, vol. 21, no. 3, pp. 58–64, Jul./Sep. 2014.

[3] A. Smolic, *et al.*, "3D video and free viewpoint video—Technologies, applications and MPEG standards," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2006, pp. 2161–2164.

[4] E. Bastug, M. Bennis, M. Medard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 110–117, Jun. 2017.

[5] B. Liu, Q. Zhu, and H. Zhu, "Delay-aware LTE WLAN aggregation in heterogeneous wireless network," *IEEE Access*, vol. 6, pp. 14544–14559, Feb. 2018.

[6] O. Galinina, A. Pyattaev, S. Andreev, M. Dohler, and Y. Koucheryavy, "5G multi-RAT LTE-WiFi ultra-dense small cells: Performance dynamics, architecture, and trends," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 6, pp. 1224–1240, Jun. 2015.

[7] J. R. Iyengar, P. D. Amer, and R. Stewart, "Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths," *IEEE/ACM Trans. Netw.*, vol. 14, no. 5, pp. 951–964, Oct. 2006.

[8] P. D. Amer. *et al.* (Mar. 2014). *Load Sharing For the Stream Control Transmission Protocol (SCTP)*. [Online]. Available: draft-tuexen-tsvwg-sctp-multipath-08.txt

[9] I. Politis, M. Tsagkaropoulos, T. Dagiuklas, and S. Kotsopoulo, "Power efficient video multipath transmission over wireless multimedia sensor networks," *Mobile Netw. Appl.*, vol. 13, nos. 3–4, pp. 274–284, Aug. 2008.

[10] W. Song and W. Zhuang, "Performance analysis of probabilistic multipath transmission of video streaming traffic over multi-radio wireless devices," *IEEE Trans. Wireless Commun.*, vol. 11, no. 4, pp. 1554–1564, Apr. 2012.

[11] M. Xing, S. Xiang, and L. Cai, "A real-time adaptive algorithm for video streaming over multiple wireless access networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 4, pp. 795–805, Apr. 2014.

[12] J. Wu, B. Cheng, C. Yuen, Y. Shang, and J. Chen, "Distortion-aware concurrent multipath transfer for mobile video streaming in heterogeneous wireless networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 4, pp. 688–701, Apr. 2015.

[13] J. Wu, C. Yuen, N.-M. Cheung, and J. Chen, "Delay-constrained high definition video transmission in heterogeneous wireless networks with multi-homed terminals," *IEEE Trans. Mobile Comput.*, vol. 15, no. 3, pp. 641–655, Mar. 2016.

[14] J. Wu, C. Yuen, N.-M. Cheung, J. Chen, and C. W. Chen, "Streaming mobile cloud gaming video over TCP with adaptive source–FEC coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 1, pp. 32–48, Jan. 2017.

[15] J. Wu, B. Cheng, M. Wang, and J. Chen, "Priority-aware FEC coding for high-definition mobile video delivery using TCP," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 1090–1106, Apr. 2017.

[16] Y. Cao, C. Xu, J. Guan, and H. Zhang, "TCP-friendly CMT-based multimedia distribution over multi-homed wireless networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2014, pp. 3028–3033.

[17] C. Xu, Z. Li, J. Li, and H. Zhang, "Cross-layer fairness-driven concurrent multipath video delivery over heterogeneous wireless networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 7, pp. 1175–1189, Jul. 2015.

[18] B. Han, F. Qian, L. Ji, and V. Gopalakrishnan, "MP-DASH: Adaptive video streaming over preference-aware multipath," in *Proc. Int. Conf. Emerg. Netw. Exp. Technol.*, New York, NY, USA, 2016, pp. 129–143.

[19] M.-F. Tsai, N. K. Chilamkurti, S. Zeadally, and A. Vinel, "Concurrent multipath transmission combining forward error correction and path interleaving for video streaming," *Comput. Commun.*, vol. 34, no. 9, pp. 1125–1136, Jun. 2011.

[20] N. Chilamkurti, J. H. Park, and N. Kumar, "Concurrent multipath transmission with forward error correction mechanism to overcome burst packet losses for delay-sensitive video streaming in wireless home networks," *Kluwer Acad. Publishers*, vol. 65, no. 2, pp. 201–220, Jul. 2013.

[21] J. Wu, C. Yuen, B. Cheng, M. Wang, and J. Chen, "Streaming high-quality mobile video with multipath TCP in heterogeneous wireless networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 9, pp. 2345–2361, Sep. 2016.

[22] J. Wu, C. Yuen, B. Cheng, M. Wang, and J. Chen, "Energy-minimized multipath video transport to mobile devices in heterogeneous wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1160–1178, May 2016.

[23] J. Wu, B. Cheng, and M. Wang, "Energy minimization for quality-constrained video with multipath TCP over heterogeneous wireless networks," in *Proc. IEEE 36th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Nara, Japan, Jun. 2016, pp. 487–496.

[24] J. Wu, B. Cheng, M. Wang, and J. Chen, "Quality-aware energy optimization in wireless video communication with multipath TCP," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 2701–2718, Oct. 2017.

[25] Y. Cao, S. Chen, Q. Liu, Y. Zuo, H. Wang, and M. Huang, "QoE-driven energy-aware multipath content delivery approach for MPT CP-based mobile phones," *China Commun.*, vol. 14, no. 2, pp. 90–103, Feb. 2017.

[26] R. Liu, M. Sheng, and W. Wu, "Energy-efficient resource allocation for heterogeneous hireless network with multi-homed user equipments," *IEEE Access*, vol. 6, pp. 14591–14601, Feb. 2018.

[27] M. Jiang and N. Ling, "Low-delay rate control for real-time H.264/AVC video coding," *IEEE Trans. Multimedia*, vol. 8, no. 3, pp. 467–477, Jun. 2006.

[28] Y. Lu, J. Xie, H. Li, and H. Cui, "GOP-level bit allocation using reverse dynamic programming," *Tsinghua Sci. Technol.*, vol. 14, no. 2, pp. 183–188, Apr. 2009.

[29] Y. Geng, X. Zhang, T. Niu, C. Zhou, and Z. Guo, "Delay-constrained rate control for real-time video streaming over wireless networks," in *Proc. Vis. Commun. Image Process. (VCIP)*, Dec. 2015, pp. 1–4.

[30] S. Zuo, I.-H. Hou, T. Liu, A. Swami, and P. Basu, "Joint rate control and scheduling for real-time wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4562–4570, Jul. 2017.

[31] K. Eymen, Y. Liu, Y. Wang, Y. Shi, C. C. Gu, and J. Lyu, "Real-time bandwidth prediction and rate adaptation for video calls over cellular networks," in *Proc. 7th Int. Conf. Multimedia Syst.*, May 2016, p. 12.

[32] *Advanced Video Coding for Generic Audiovisual Services*, document ITU-T H.264, ISO/IEC 14496-10, Nov. 2007.

[33] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video*, document H.262, ISO/IEC, 1993.

[34] H. Lee, T. Choi, and S. Sull, "Frame layer CBR encoding to reduce large picture quality fluctuation," *IEEE Trans. Consum. Electron.*, vol. 58, no. 3, pp. 1031–1037, Aug. 2012.

[35] Z. Chen and Y. Reznik, "Analysis of video codec buffer and delay under time-varying channel," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Nov. 2012, pp. 1–6.

[36] Y. Shuai and T. Herfet, "On stabilizing buffer dynamics for adaptive video streaming with a small buffering delay," in *Proc. 14th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2017, pp. 435–440.

[37] P. Zhao, W. Yu, X. Yang, D. Meng, and L. Wang, "Buffer data-driven adaptation of mobile video streaming over heterogeneous wireless networks," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3430–3441, Oct. 2018.

[38] D. Yu-ning, W. Zheng, and C. Liu, "A joint rate and buffer control scheme for video transmission over LTE wireless networks," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Mar. 2018, pp. 26–30.

[39] S. Kim and Y. Won, "Frame rate control buffer management technique for high-quality real-time video conferencing system," in *Proc. Adv. Comput. Sci. Ubiquitous Comput.*, 2017, pp. 777–783.

[40] W. Yu, W. Rhee, S. Boyd, and J. M. Cioffi, "Iterative water-filling for gaussian vector multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 1, pp. 145–152, Jan. 2004.

[41] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," in *Proc. 3rd IEEE Int. Conf. Image Process.*, Sep. 1996, pp. 73–76.

[42] S. Vivienne, M. Budagavi, and G. J. Sullivan, *High Efficiency Video Coding (HEVC): Algorithms and Architectures* (Integrated Circuits and Systems), 2014.

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

[44] D. L. Mills. (Sep. 1985). *Network Working Group, Network Time Protocol (NTP)*. [Online]. Available: https://tools.ietf.org/html/rfc958

[45] V. J. Ribeiro, R. H. Riedi, R. G. Baraniuk, J. Navratil, and L. Cottrell, "pathChirp: Efficient available bandwidth estimation for network paths," in *Proc. Passive Active Meas. Workshop*, 2003, pp. 1–11.

[46] (2018). *HoloWan Simulator*. [Online]. Available: http://www.msytest.cn/

[47] D. Newman, *Benchmarking Terminology for Firewall Performance*, document RFC 2647, Aug. 1999. [Online]. Available: https://tools.ietf.org/html/rfc2647

FENG CHEN was born in 1984. He received the M.S. degree in communication engineering from the Guilin University of Electronic Technology, China, in 2009, and the Ph.D. degree in signal and information processing from the Beijing University of Posts and Telecommunications (BUPT), China, in 2014. He is currently an Assistant Professor with the College of Physics and Information Engineering, Fuzhou University, China. His current research interest includes radio resource management, heterogeneous networks, and multimedia communication.

JIE ZHANG received the B.S. degree in communication engineering from Jimei University, Xiamen, China, in 2016. She is currently pursuing the master's degree in signal and information processing with Fuzhou University, Fuzhou, China. Her current research interest is real-time video concurrent multipath transmission in heterogeneous wireless networks.

ZHIFENG CHEN (SM'18) received the Ph.D. degree in electrical and computer engineering from the University of Florida, USA, in 2010. He is currently a Professor with the College of Physics and Information Engineering, Fuzhou University, China. His research interests include computer vision, machine learning, video coding, and video transmission. He was invited as a Guest Editor for E-Letter of the IEEE Communications Society Technical Committee on Multimedia Communications, a member of the Technical Program Committee for eight international conferences, and a Reviewer for more than 15 prestigious journals and conferences. He was awarded the Best Reviewer for IEEE VCIP, 2013.

JIYAN WU received the bachelor's degree from the North China University of Technology, the master's degree from the China University of Mining and Technology, and the Ph.D. degree in computer science and technology from the Beijing University of Technology. He was a Postdoctoral Research Fellow with Nanyang Technological University, Singapore, and a Software Developer (C++) from with Visiosoft Technologies Co., Ltd., China. He is currently a Senior Researcher with the College of Physics and Information Engineering, Fuzhou University, and also an Adjunct Research Scientist with the Advanced Digital Science Center, Singapore. His research interests include real-time video communication, cloud/edge computing, heterogeneous wireless networks, and embedded systems.

NAM LING (S'88–M'90–SM'99–F'08) received the B.Eng. degree from the National University of Singapore, Singapore, in 1981, and the M.S. and Ph.D. degrees from the University of Louisiana, Lafayette, LA, USA, in 1985 and 1989, respectively.

From 2002 to 2010, he was the Associate Dean of the School of Engineering, Santa Clara University, Santa Clara, CA, USA, where he is currently the Sanfilippo Family Chair Professor and the Chair of the Department of Computer Engineering. He is/was also a Consulting Professor with the National University of Singapore, a Guest Professor with Tianjin University, Tianjin, China, a Guest Professor with Shanghai Jiao Tong University, Shanghai, China, a Cuiying Chair Professor with Lanzhou University, Lanzhou, China, a Chair Professor and a Minjiang Scholar with Fuzhou University, Fuzhou, China, a Distinguished Professor with the Xi'an University of Posts and Telecommunications, Xi'an, China, a Guest Professor with the Zhongyuan University of Technology, Zhengzhou, China, and an Outstanding Overseas Scholar with the Shanghai University of Electric Power, Shanghai. He has authored or co-authored over 210 publications, and he has adopted seven standard contributions. He has filed over 20 U.S. patents and 14 patents have been granted so far.

Dr. Ling is an IEEE Fellow for his contributions to video coding algorithms and architectures. He was named as an IEEE Distinguished Lecturer twice, and he was also an APSIPA Distinguished Lecturer. He has received the IEEE ICCE Best Paper Award (First Place) and the IEEE Umedia Best Paper Award twice. He was a recipient of six awards from Santa Clara University, four at the University level (outstanding achievement, recent achievement in scholarship, president's recognition, and sustained excellence in scholarship), and two at the School/College Level (researcher of the year and teaching excellence). He was a Keynote Speaker of the IEEE APCCAS, VCVP (twice), JCPC, the IEEE ICAST, the IEEE ICIEA, IET FC Umedia, the IEEE Umedia, the IEEE ICCIT, and the Workshop at XUPT (twice), and a Distinguished Speaker of the IEEE ICIEA. He has served as the General Chair/Co-Chair for the IEEE Hot Chips, VCVP (twice), the IEEE ICME, the IEEE Umedia (six times), and the IEEE SiPS. He was the Honorary Co-Chair for the IEEE Umedia 2017. He has also served as the Technical Program Co-Chair of the IEEE ISCAS, APSIPA ASC, the IEEE APCCAS, the IEEE SiPS (twice), DCV, and the IEEE VCIP. He was the Technical Committee Chair of the IEEE CASCOM TC and the IEEE TCMM. He has served as a Guest Editor or Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-I: REGULAR PAPERS, the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, the IEEE ACCESS, Springer JSPS, and Springer MSSP.