A Novel Scheduling Strategy for MMT-based Multipath Video Streaming

Samira Afzal*, Vanessa Testoni†, Jean Felipe F. de Oliveira†,
Christian Esteve Rothenberg*, Prakash Kolan‡, Imed Bouazizi‡
*School of Electrical and Computer Engineering (FEEC), University of Campinas (Unicamp), Campinas, São Paulo, Brazil
†Samsung Research Brazil (SRBR), Campinas, São Paulo, Brazil
‡Samsung Research America (SRA), Dallas, Texas, USA

Abstract*—Bandwidth constraints and high end-to-end delays are real challenges for achieving and sustaining high quality mobile video streaming services. Diverse multipath transmission techniques are being investigated as possible solutions, since recent developments have enabled mobile devices users to receive video data simultaneously over multiple interfaces (e.g., LTE and WiFi). While some multipath protocols have been recently standardized for this purpose (e.g., MPTCP), being network layer protocols they cannot properly handle challenging transmission scenarios subject to packet losses and congestion, such as lossy wireless channels. In this work, we adopt the MPEG Media Transport (MMT) protocol to propose an improvement for mobile multipath video streaming solutions. MMT is an application layer protocol with inherent hybrid media delivery properties. We propose a novel path-and-content-aware scheduling strategy for MMT by means of full cooperation between network metrics and video content features. Our strategy provides better models to adaptively cope with unstable communication channel conditions and to improve the final user quality of experience (QoE). For the experimental evaluation, we used NS3-DCE to simulate a realistic multipath network scenario which includes channel error models and background traffic. Results for two video sequences are presented in terms of PSNR, SSIM, goodput, delay and packet loss rates. When compared with a simple scheduling strategy for the traditional multipath MMT, our approach yields significant packet loss rate reductions (~ 90%) and video quality improvements of around 12 dB for PSNR and 0.15 for SSIM.

***Index Terms*—Video streaming, MMT protocol, multipath streaming, path-and-content-aware scheduling.**

I. INTRODUCTION

With the fast development of media technologies, applications such as on-demand-video, video conferences and online cloud gaming have been responsible for an growing amount of network traffic. According to the annual Cisco's report [1], it would take an individual more than 5 million years to watch the amount of video that will cross global IP networks per month in 2021.

End users always expect a high quality video streaming service, regardless of the network situation, and it is well-known that achieving it requires mainly high bandwidth and low transmission delay. Ensuring these prerequisites is a challenge, especially on lossy wireless channels. One candidate solution is to exploit multipath video streaming strategies by taking advantage of the multiple network interfaces that are currently available in most mobile devices. Since the user can be connected to more than one network, these strategies enable better coverage and, therefore, more stable network services. The network congestion is also relieved by the aggregation of the bandwidth available over multiple paths.

Several efforts have been done regarding multipath video streaming [2], [3]. Two well-known multipath protocols are the Stream Control Transmission Protocol (SCTP) [4] and the Multipath TCP (MPTCP) [5]. Both protocols are implemented in the transport network layer without access to video content features available on the application network layer. Some attempts to improve multipath streaming and scheduling strategy by extending MPTCP or SCTP protocols can be found in [6]–[9]. In addition, some works adapting the Dynamic Adaptive Streaming over HTTP (MPEG-DASH) protocol [10] for multipath streaming can be found in [9], [11].

There is also the MPRTP draft standard [12] with enabled multipath capability for Real-time Transport Protocol (RTP) [13]. Pakulova et al. [14] proposed a solution using MPRTP protocol, performing video bit rate adaptation considering throughput and packet loss rate of each path. However, in this solution, video packet content priority is not considered. Singh et al. [15] proposed a similar solution to our work supporting adaptive video traffic splitting based on network condition and enabling content-aware capabilities. Some differences in relation to our work are the use of packet retransmission in MPRTP and our proposed scheduling strategy, which is the subject of this paper and will be detailed in the remainder of this text.

In this work, we chose to exploit the advantages of multipath streaming by adopting the MPEG Media Transport (MMT) protocol [16]–[18]. We propose a novel scheduling strategy which considers both path characteristics (path-aware) and video features (content-aware). Our path-and-content-aware scheduling strategy can improve quality of experience (QoE) by increasing goodput, decreasing packet losses and reducing end-to-end delay. To the best of our knowledge, this is the first attempt to improve the MMT standard by adding multipath scheduling strategies. It is important to note, however, that our approach does not require any change in the protocol itself since the scheduler can be implemented as part of the client/server applications.

We believe that MMT is an appropriate protocol for ex-

978-1-5386-4727-1/18/$31.00

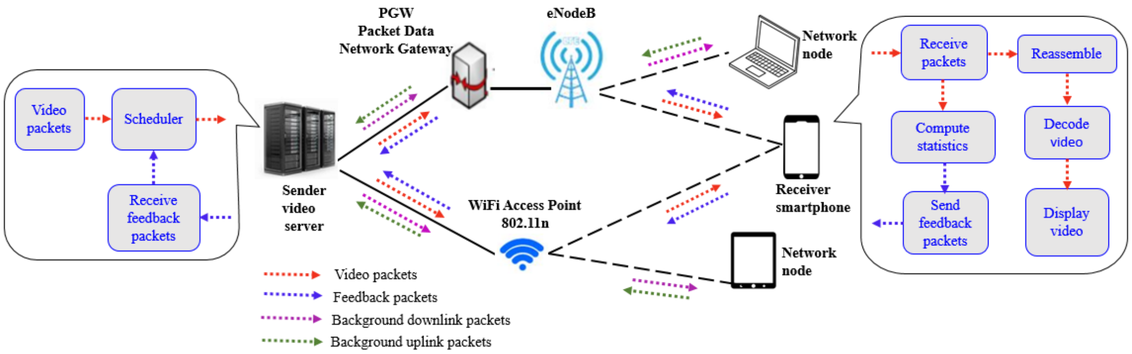

Fig. 1. Overview of the considered multipath mobile video streaming scenario.

ploiting multipath streaming because it is a multimedia application layer protocol with the inherent ability to transmit video in heterogeneous network environments. Actually, the capability of hybrid media delivery is one of the main MMT properties. Hybrid media delivery refers to the combination of delivered media components over different types of network. For example, it could be one broadcast channel and one broadband, or it could be two broadband channels. MMT was standardized in 2013 and has already been adopted by several other standards as a replacement solution for the old MPEG-2 TS protocol [19], [20].

The structure of this paper is organized as follows. In Section II, the system model and the proposed path-and-content-aware strategy are explained. Section III describes our network simulation scenario and provides the performance evaluation. Finally, in Section IV, we conclude the paper and discuss future work topics.

II. PATH AND CONTENT AWARE SCHEDULING PROPOSAL

An overview of the multipath video streaming scenario considered in this work is presented in Figure 1. We define a unicast video transmission system employing the Multipath User Datagram Protocol (MPUDP) over heterogeneous wireless networks (WiFi and LTE). The sender follows a path-and-content-aware strategy defined by its scheduler component to allocate the data through the multiple transmission paths. The scheduler monitors the current conditions of each path and verifies the content of each MMT video packet to select the best path for each packet to be transmitted.

Since our scenario considers a heterogeneous network context, it is important to properly allocate transmission bit rates that cope with each path capacity and its current conditions in order to avoid either congestion or underutilization of network resources. The adaptive traffic split scheme proposed here is described in Subsection II-A.

Objective metrics, such as goodput, average delay, number of lost packets, and jitter, are periodically computed at the receiver side in order to monitor the quality of the transmission paths. In this work, these metrics are computed and sent as feedback messages to the sender every 0.5 seconds by using the following two signaling messages defined in MMT standard [16]: Reception Quality Feedback (RQF) and Network Abstraction for Media Feedback (NAMF).

Considering the relevance of the feedback information for the proper scheduling process, it is necessary to send it over the most qualified path. Therefore, the receiver selects the best path to return feedback packets based on the last computed objective metrics. After a feedback interval time, at the sender side, the scheduler uses the received metrics to adaptively categorize each path condition as good, mild or bad by following the method detailed in Subsection II-B. If the path condition is estimated as mild or bad, the possibility of loosing video packets is high, either because they will be dropped by the network itself or because they will be considered as overdue packets when arriving at the receiver side. Therefore, the scheduler also applies the discard strategy described in Subsection II-C to help reducing congestion by avoiding sending packets that would probably be lost.

The scheduler also considers the content of each video packet to better protect I frames and the closest n P frames, named as near-I (NI) frames in this work. The reason for protecting NI frames is based on the fact that errors on these first P frames of the group of pictures (GOP) have a higher impact on the perceived quality of experience [21]. Our proposed content-aware strategy follows the rules described in Subsection II-D.

A. Adaptive Video Traffic Split

We propose to split the video traffic based on a goodput-division-delay (GDD) metric dynamically computed by the scheduler after receiving a feedback packet. The GDD for each path is calculated as $GDD_p = \frac{gp_p}{d_p}$, where gp_p is the goodput of path p measured in [kbps] and d_p is the average one-way delay of path p measured in [s]. The bit rate split factor for each path (λ_p) is then computed as

$$\lambda_p = \frac{GDD_p}{\sum_{i=1}^{N} GDD_p}, \tag{1}$$

where N is the total number of transmission paths. As an initial estimation, when feedback packets were not yet received by the scheduler, λ_p is computed as

$$\lambda_p = \frac{\frac{bw_p}{delay_p}}{\sum_{i=1}^{N} \frac{bw_p}{delay_p}}, \tag{2}$$

where bw_p is the inherent maximum path bandwidth and $delay_p$ is the minimum path delay.

B. Estimation of Path Condition

We propose to model the path condition estimation problem as a three-state Markov model where each state represents one path condition: Good Condition (**GC**), Mild Condition (**MC**) and Bad Condition (**BC**). A matrix P of transition probabilities among the three states is computed and periodically updated by the scheduler. A matrix C is also kept to store the number of transitions from each state i to state j (c_{ij}). Following [22], the elements of matrix P are computed by the following equation:

$$p_{ij} = \frac{c_{ij} + 1}{\sum_{j=1}^{N} c_{ij} + N}, \tag{3}$$

where p_{ij} is the transition probability from i to j.

In order to define the path condition state, two predefined thresholds are used: T_d for one-way delay and T_l for packet loss rate. T_d was set as 50 milliseconds following recommendations in [23], where this is the maximum limit delay for achieving high quality multipath HD video transmission in heterogeneous wireless networks. For the definition of T_l, we were inspired by the work in [24], which specifies a multipath streaming scheme (EMS) with FEC (Forward Error Correction). The work states that, with H.264 encoding, the packets loss rate should be less than 1% in order to ensure high quality real-time live streaming. Since FEC is not applied in this work to MMT packets, this limit was slightly extended and T_l was set as 2%.

The following two metrics specified in [25] were also computed and used in this work:

$$d_{p,wma,cur} = \frac{31}{32} \cdot d_{p,wma,pre} + \frac{1}{32} \cdot d_p \tag{4}$$

$$\sigma_{d_{p,cur}} = \frac{15}{16} \cdot \sigma_{d_{p,pre}} + \frac{1}{16} \cdot |d_p - d_{p,wma,cur}| \tag{5}$$

where $d_{p,wma,cur}$ is the current weighted moving average of one-way delay of path p, $d_{p,wma,pre}$ is the previous weighted moving average of one-way delay of path p, $\sigma_{d_{p,cur}}$ is the current standard deviation of one-way delay of path p and $\sigma d_{p,prev}$ is the previous standard deviation of one-way delay of path p.

The two thresholds (T_d and T_l) and the two computed metrics ($d_{p,wma,cur}$ and $\sigma_{d_{p,cur}}$) are then combined in the following way:

- Path is in **GC** state if $D_p \leq T_d$ && $L_p \leq T_l$, where D_p and L_p are the highest one-way delay of path p and packet loss rate of path p, respectively, for a single feedback interval time;
- Path is in **MC** state if $D_p \leq T_d$ && $L_p > T_l$;
- Path is in **BC** state if $D_p > T_d \,\|\, d_p > d_{p,wma,cur} + \frac{\sigma_{d_{p,cur}}}{2}$.

C. Discard Strategy

If the path condition state is estimated as **MC** or **BC**, a discard strategy is applied by the scheduler to avoid increasing network congestion by not even sending packets that will be probably either overdue or dropped. As part of the strategy, the transition probabilities computed according to Eq. (3) are used to reduce the path bit rate split factor λ_p in the following way:

- if the path is in **MC** it is important to consider the path history in order to verify if there is a higher probability of moving to **GC** or to **BC**. The last computed objective metrics are compared with the metrics received in the previous feedback message. If the number of lost packets, jitter and delay have increased, then the probability of moving to **BC** is higher and λ_p is updated as $\lambda_p = \lambda_p \, .(1-p_{MB})$, where p_{MB} is the probability of transition from **MC** to **BC**. Otherwise, the path conditions are improving and no packet will be discarded;
- if the path is in **BC**, then $\lambda_p = \lambda_p \, . \, (1 - p_{BB})$, where p_{BB} is the probability of being in **BC** and staying in **BC** state.

D. Adding Content-aware Protection

In addition to the path-aware scheduling strategy detailed in Subsections II-A, II-B and II-C, we also define a content-aware scheme to better protect I and near-I (NI) frame packets in this work. Our content-aware scheme does not discard any I or NI packet and employs packets duplication and/or rerouting according to the following rules specified for the scenario with two transmission paths ($N = 2$):

- if one path is in **GC** and the other path is in **MC** or **BC**, then all I packets will be sent only through the path in **GC**;
- if one path is in **MC** and the other path is in **BC** or both paths are in **BC**, then all I packets will be duplicated and sent through both paths;
- if both paths are in **GC** or in **MC**, then no packet will be duplicated and the bit rate split factors computed according to Eq. (1) for each path will be used.

The same rules are applied for NI frame packets, except they are never duplicated. The protection scheme is not applied to P frame packets. Therefore, they are transmitted according to all the rules specified in previous subsections, including the discard strategy. It must be highlighted that our path-and-content-aware strategy does not discard any packet containing I or NI frame data. In other words, the proposed discard strategy applies to P frame packets only.

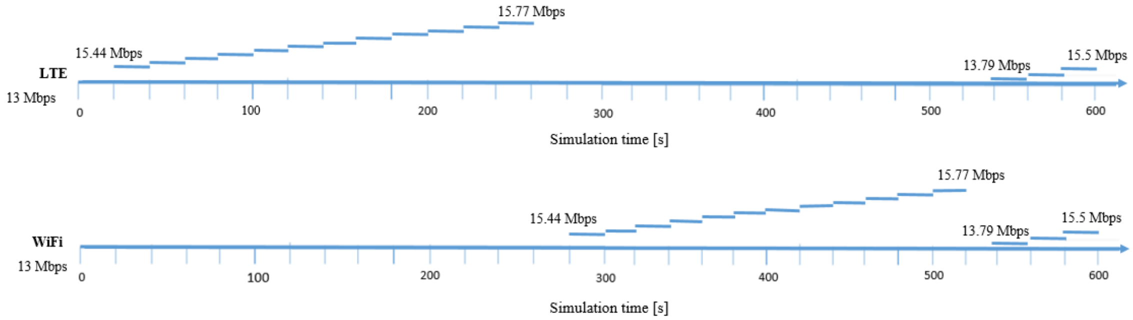

Fig. 2. Variable background traffic setup to first simulate congestion separately in each path and then simultaneous congestion in both paths.

TABLE I
PACKETS DISTRIBUTION ACCORDING TO FRAME TYPE.

Video sequence	**I packets**	**NI packets**	**P packets**
Elephants Dream	24.36%	11.29%	64.35%
Big Buck Bunny	38.23%	10.92%	50.85%

III. EXPERIMENTAL EVALUATION

A. Methodology

The experiments of the proposed path-and-content-aware scheduling strategy are carried for the well-known cartoon sequences named *Elephants Dream* and *Big Buck Bunny*. Their resolutions are of 1920×1080 and the total number of frames is approximately the same: 15,691 frames for *Elephants Dream* and 14,315 frames for *Big Buck Bunny*. The H.264/AVC JM Reference Software [26] was used as the encoding tool and the MP4 fragmentation procedures were done by the GPAC MP4BOX [27] tool. Decoding and error concealment were performed with FFMPEG [28]. The employed GOP structure is IPPPP...P and the GOP size is of 16 frames.

In order to compare results in the same network simulation scenario, both sequences were encoded with the same source bit rate (4 Mbps). Table I shows the distribution of packets and one can see that *Big Buck Bunny* has more I packets and consequently less P packets than *Elephants Dream*. This difference is due to higher amount of texture, details and action in *Big Buck Bunny*. Only the initial 2 P frames in the GOP were considered as near-I (NI) frames. Therefore, the remaining 13 frames in each GOP are regular P frames. Table I also shows the distribution of NI and P packets.

A NS3-DCE [29] model is implemented to simulate the proposed strategy in a realistic network scenario. Our multipath simulation setup comprises of two wireless networks implemented by the LTE and WiFi modules available in the NS-3 simulation library. The different specifications and heterogeneity between LTE and WiFi are among the big challenges of this work for properly splitting video traffic and achieving load balancing. For the LTE path, based on [30], bw_p and $delay_p$ were defined, respectively, as 18.3 Mbps and 15 milliseconds (ms). The 802.11n/5GHz model was chosen for the WiFi path. Therefore, bw_p and $delay_p$ were defined for the WiFi path, respectively, as 54 Mbps and 10 milliseconds.

Constant uplink and variable downlink background traffics were added in the NS3 simulation, as illustrated in Figure 1, to stress out the proposed scheduling strategy in different network congestion situations. The downlink background traffic is generated by the server and initially set as 13 Mbps for both paths. On the other hand, the uplink background traffic is generated by the network nodes and set as 2 Mbps, in accordance to real network scenarios where the uplink traffic is smaller than the downlink traffic.

In our simulation, the uplink background traffic is kept constant, but the downlink background traffic is periodically increased separately for each path, as illustrated in Figure 2. In the first part of the simulation (approximately until 250 ms), the background traffic of only LTE path is increased while the background traffic of the WiFi is kept constant. In the second part (approximately until 500 ms), the opposite behaviour is simulated and the background traffic of the WiFi path is increased while the background traffic of the LTE is kept constant. Finally, after 500ms, background traffic is slightly increased in both paths to simulate simultaneous congestion in LTE and WiFi.

In order to turn the simulation setup more real, the NS3 channel random error model was employed to capture the effects of noisy wireless channels. Loss rate values were set as 1% for the LTE path and 0.1% for the WiFi path, based on the research in [31].

B. Performance Evaluation

The following objective metrics were computed to evaluate the performance of the proposed path-and-content-aware scheduling strategy: goodput, packet loss rate, PSNR and SSIM. The results were compared with a simple scheduling strategy where packets were evenly split in both transmission channels. It was not possible to compare our results with other more sophisticated approaches because we were not able to find other works proposing MMT-based multipath scheduling improvements. However, it is important to note that, even

TABLE II

COMPARISON RESULTS BETWEEN DIFFERENT SCHEDULING STRATEGIES FOR *Elephants Dream* AND *Big Buck Bunny* VIDEO SEQUENCES.

Video sequence	**Scheduling strategy**	**Total packet loss rate**	**I and NI frame packet loss rate**	**PSNR**	**SSIM**	φ_{LTE}	φ_{WiFi}
Elephants Dream	Path-and-content-aware	**2.6%**	**0.8%**	**39.56 dB**	**0.942**	32%	23%
	Evenly split	23.48%	8.5%	27.24 dB	0.768		
Big Buck Bunny	Path-and-content-aware	**4.2%**	**1.6%**	**32.54 dB**	**0.893**	34%	32%
	Evenly split	27.4%	12.89%	21.13 dB	0.749		

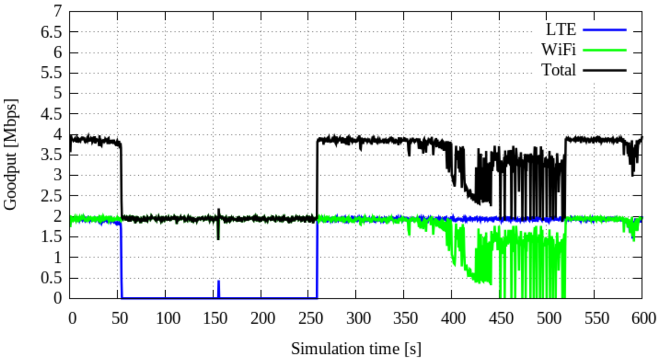

Fig. 3. LTE, WiFi and total (joint) goodput for *Elephants Dream* packets evenly split through WiFi and LTE transmission channels.

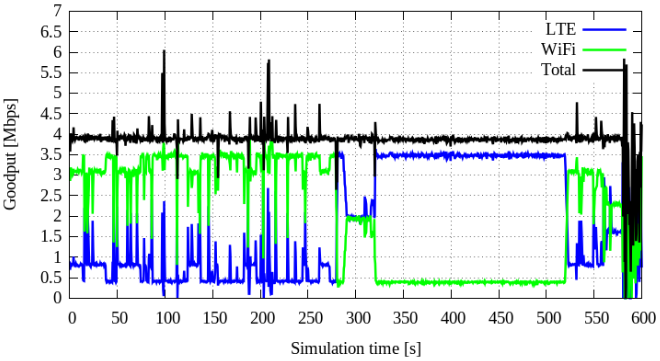

Fig. 4. LTE, WiFi and total (joint) goodput for *Elephants Dream* packets transmitted according to the proposed path-and-content-aware scheduling strategy.

though evenly splitting packets is a simple scheduling strategy, it can take advantage of the network multipath capabilities and increase the total achieved goodput, as can be seen in Figure 3 for the transmission of the video sequence *Elephants Dream*.

One can note in Figure 3 that, at the initial part of the simulation, where there is no congestion, video traffic is equally split between LTE and WiFi. Therefore, each channel has a goodput of 2 Mbps and the total goodput is of 4 Mbps. Then, when LTE gets heavily congested, its goodput sharply decreases due to packet losses, and the total goodput (2 Mbps) achieved in this period is only due to packets transmitted over WiFi. In the second part of the simulation, LTE recovers from congestion and then WiFi gets congested. However, since WiFi inherent capacity is higher than LTE, congestion is better handled and its goodput is only slowly decreased while the effect on the total goodput is not so noticeable.

Figure 4 illustrates that the total achieved goodput is higher and more stable when our proposed scheduling strategy is applied instead of the evenly split distribution. One can note that load balancing is clearly achieved between both paths and the higher inherent capacity of the WiFi path is exploited. In addition, congestion (first in LTE and then in WiFi) is properly handled by the scheduler by switching traffic among paths and keeping a stable total goodput of 4 Mbps through all simulation, except for the last part where both paths get congested. The goodput decrease in this last part is not only due to congestion, but also to packets lost due to the discard strategy applied by the scheduler to avoid further congestion.

Table II shows a comparison between the packet loss rates for the scenarios simulated in Figures 3 and 4 with *Elephants Dream*. Results are also provided for *Big Buck Bunny*, considering the same simulation scenarios. One can see in Table II that, due to differences of the video sequences previously mentioned and shown in Table I, *Big Buck Bunny* has more losses than *Elephants Dream* in all compared conditions. However, one common result for the two sequences and all compared conditions is the significant packet loss rate reduction, varying from **85%** to around **91%**, achieved by the path-and-content-aware scheduling strategy when compared to the evenly split scheduling strategy. For instance, the total packet loss rate of *Elephants Dream* is reduced from 23.48% to only 2.6%, which corresponds to a 88.93% rate reduction.

Results in Table II also show the effectiveness of the proposed scheduling strategy to better protect I and NI frame packets. For both sequences, the percentage of lost I and NI frame packets over the total number of lost packets decreases. For instance, with our proposed scheduling strategy, I and NI losses for *Elephants Dream* decrease to only 0.8%, which corresponds to 30.76% of the total losses (2.6%). When evenly split scheduling strategy is applied, I and NI losses correspond to 36.20% of total losses. Therefore, the rate is reduced from 36.20% to 30.76%. Similar results are obtained for *Big Buck Bunny*, where the rate is reduced from 46.71% to only 38.09%.

Additionally, Table II shows the reduction of average one-way delay of non-overdue packets for LTE (φ_{LTE}) and WiFi (φ_{WiFi}). For instance, the Elephants Dream sequence video has shown a reduction of 32% and 23%, respectively, for LTE and WiFi. The results clearly indicate, for both simulated scenarios, that our proposed path-and-content-aware strategy efficiently enables a better adjustment to network conditions by balancing the bit rate distribution and the discard strategy.

Regarding the objective video quality metrics results, the PSNR and SSIM values of Table II attest to our objective of improving the QoE of end users by employing our scheduling strategy. One can see that PSNR and SSIM gains of, respectively, **12.34 dB** and **0.174** can be achieved for *Elephants Dream* while similar gains of **11.41 dB** and **0.144** can be achieved for *Big Buck Bunny*.

Under the simulated network conditions, as shown in Table II, the resulting QoE would be very low, almost completely degrading the whole sequence with PSNR values as low as 21.13 dB. Even in this challenging scenario, our proposed strategy was able to keep the QoE in higher levels while optimizing the total network goodput.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a novel scheduling strategy as a multipath MMT solution with the ability to adapt to varying network path conditions taking into account the characteristics of the video content. The main advantage of this path-and-content-aware approach is the ability to improve the selection of the best path for transmitting each video packet.

Our approach successfully proved its efficiency in a simulated network scenario, ensuring high levels of PSNR and SSIM in conditions where a simple multipath scheduling solution results in a totally degraded video experience due to the high number of lost packets. It is important to highlight that, since we leveraged the feedback signaling mechanisms defined in the MMT standard, this work is a candidate solution for future improvements of MMT standardization efforts.

As future work, we consider adding a Forward Error Correction (FEC) stage exploiting what is already specified as the Application Layer FEC of the MMT standard. Improvements of the current discard and content-aware strategies are also considered. Moreover, since we were not able to find any multipath MMT approach in the reviewed literature, we plan to compare our proposed strategy with alternative multipath solutions, such as MPRTP [12], [15], in addition to experimental evaluation using the Mininet-WiFi emulator [32].

ACKNOWLEDGMENT

This work was partially supported by Samsung Research Brazil and CNPq (Brazilian National Council for Scientific and Technological Development).

REFERENCES

[1] “Cisco Visual Networking Index: Forecast and Methodology 2016-2021,” 2016.

[2] J. Chakareski, S. Han, and B. Girod, “Layered coding vs. multiple descriptions for video streaming over multiple paths,” *Multimedia Systems*, vol. 10, no. 4, pp. 275–285, 2005.

[3] J. Chakareski and B. Girod, “Rate-distortion optimized packet scheduling and routing for media streaming with path diversity,” *Data Compression Conference*, pp. 203–212, 2003.

[4] R. Stewart, “Stream control transmission protocol,” *RFC 4960*, 2007.

[5] A. Ford, C. Raiciu, S. Barre, and Louvain, “TCP Extensions for Multipath Operation with Multiple Addresses,” *RFC 6824*, 2009.

[6] C. Xu, T. Liu, J. Guan, H. Zhang, and G.-M. Muntean, “CMT-QA: Quality-aware adaptive concurrent multipath data transfer in heterogeneous wireless networks,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2193–2205, 2013.

[7] 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 Transactions on Mobile Computing*, vol. 14, no. 4, pp. 688–701, 2015.

[8] J. Wu, C. Yuen, B. Cheng, M. Wang, and J. Chen, “Streaming high-quality mobile video with multipath TCP in heterogeneous wireless networks,” *IEEE Transactions on Mobile Computing*, vol. 15, no. 9, pp. 2345–2361, 2016.

[9] B. Han, F. Qian, L. Ji, V. Gopalakrishnan, and N. Bedminster, “MP-DASH: Adaptive Video Streaming Over Preference-Aware Multipath,” *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies*, pp. 129–143, 2016.

[10] “MPEG systems technologies - Part 6: Dynamic adaptive streaming over HTTP (DASH) ,” *ISO/IEC FCD 23001-6*, January 2011.

[11] P. Houzé, E. Mory, G. Texier, and G. Simon, “Applicative-layer multipath for low-latency adaptive live streaming,” *IEEE International Conference on Communications (ICC)*, pp. 1–7, 2016.

[12] V. Singh, T. Karkkainen, J. Ott, and S. Ahsan, “Multipath RTP (MPRTP),” *IETF Draft, draft-singh-avtcore-mprtp*, 2012.

[13] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A Transport Protocol for Rea-l Time Applications,” *RFC 3550*, 2003.

[14] E. Pakulova, K. Miller, and A. Wolisz, “Adaptive low-delay video streaming in heterogeneous wireless networks using mprtp,” in *Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International*. IEEE, 2017, pp. 14–19.

[15] V. Singh, S. Ahsan, and J. Ott, “MPRTP: multipath considerations for real-time media,” *Proceedings of the 4th ACM Multimedia Systems Conference*, pp. 190–201, 2013.

[16] “High efficiency coding and media delivery in heterogeneous environments - MPEG-H Part 1: MPEG Media Transport (MMT) ,” *ISO/IEC 23008-1*, 2014.

[17] P. Kolan and I. Bouazizi, “Method and apparatus for multipath media delivery,” Dec. 22 2016, uS Patent App. 15/001,018. [Online]. Available: https://www.google.com/patents/US20160373342

[18] “MMT Enhancements for mobile environments,” *ISO/IEC 23008-1:2017/DAmd 2*, 2017.

[19] S. J. Bae, “Method for configuring and transmitting m-unit,” Apr. 18 2013, uS Patent App. 13/651,191. [Online]. Available: http://www.google.sr/patents/US20130094594

[20] Y. Ye, Y. He, Y.-K. Wang *et al.*, “SHVC the Scalable Extensions of HEVC and Its Applications,” *ZTE Communications*, vol. 14, no. 1, 2016.

[21] Á. Huszák and S. Imre, “Analysing GOP structure and packet loss effects on error propagation in MPEG-4 video streams,” *Proceedings of the 4th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pp. 1–5, 2010.

[22] M. Xing, S. Xiang, and L. Cai, “A real-time adaptive algorithm for video streaming over multiple wireless access networks,” *IEEE Journal on Selected Areas in communications*, vol. 32, no. 4, pp. 795–805, 2014.

[23] 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 Transactions on Mobile Computing*, vol. 15, no. 3, pp. 641–655, 2016.

[24] A. L. Chow, H. Yang, C. H. Xia, M. Kim, Z. Liu, and H. Lei, “EMS: Encoded multipath streaming for real-time live streaming applications,” *17th IEEE International Conference on Network Protocols (ICNP)*, pp. 233–243, 2009.

[25] S. Cen, P. C. Cosman, and G. M. Voelker, “End-to-end differentiation of congestion and wireless losses,” *IEEE/ACM Transactions on networking*, vol. 11, no. 5, pp. 703–717, 2003.

[26] “H.264/AVC JM Reference Software, Version 19.0,” 2015.

[27] “GPAC MP4BOX, Version 0.5.1,” 2015.

[28] “FFmpeg, Version 3.4.1,” 2017.

[29] “NS3 Direct Code Execution (DCE), Version 1.9 ,” 2016.

[30] “OpenSignal,” https://opensignal.com/reports/2017/02/global-state-of-the-mobile-network, 2017, accessed:15-JAN-2018.

[31] Y.-C. Chen, Y.-s. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley, “A measurement-based study of multipath tcp performance over wireless networks,” *Proceedings of the 2013 conference on Internet measurement conference*, pp. 455–468, 2013.

[32] Ramon Fontes, Samira Afzal, Samuel Brito, Mateus Santos and Christian Esteve Rothenberg. “Mininet-WiFi: Emulating Software-Defined Wireless Networks,” *In 2nd International Workshop on Management of SDN and NFV Systems 2015*, Barcelona, Spain, Nov., 2015.