SDN-based in-network early QoE prediction for stable route selection on multi-path network

Shumpei Shimokawa, Yuzo Taenaka, Kazuya Tsukamoto, Myung Lee

Abstract As QoE is useful to uniformly handle many kinds of application flows, we have been tackling QoE-oriented network resource management based on SDN technology. Toward this goal, our previous study proposed a QoE measurement method for on-going streaming flows. However, the standard QoE calculation model requires at least 8 seconds for collecting the flow information. In this study, we tackle early QoE prediction on a SDN-enabled multi-path network. To predict video QoE as soon as possible, we exploit not only packet loss rate measured regularly but also the number of packet transmissions by short-period measurement at the flow arrival. Finally, through experiments, we demonstrated that QoE of all flows can be maximized by selecting an appropriate route based on the predicted QoE.

1 Introduction

Network resource management, such as route selection, is one of key technologies to make multiple applications coexist in a network with reliable performance. Our previous study tackled the efficient resource utilization to keep required throughput for every flows [1]. However, as many kinds of applications coexist in a realistic network, throughput is not the best performance metric for some applications. To keep application performance of any kind of application flows, Quality of Experience (QoE) could be a common performance metric for them.

To use QoE as a metric for a route selection, we need to track QoE for every flows by exploiting in-network information. Our previous study proposed SDN-based in-network QoE measurement method for video streaming flows [2]. However, the standard QoE calculation model requires at least 8 seconds for collecting flow information from the network, and thus QoE-based route selection is quite difficult at the time of a new flow arrival. Although there is a very few information we can

Shunpei Shimokawa
Kyushu Institute of Technology, Japan, e-mail: shimokawas@infonet.cse.kyutech.ac.jp

Yuzo Taenaka
Nara Institute of Science and Technology, Japan, e-mail: yuzo@is.naist.jp

Kazuya Tsukamoto
Kyushu Institute of Technology, Japan, e-mail: tsukamoto@cse.kyutech.ac.jp

Myung Lee
CUNY, City College, USA, e-mail: mlee@ccny.cuny.edu

obtain for an arrival flow, predicting QoE and accordingly handling the flow are essential to keep good QoE. Therefore, we propose an early QoE prediction method on SDN-enabled multi-path network. Specifically, we predict video QoE metric by exploiting not only packet loss rate measured regularly but also the number of arrival bytes for very short period at the OFC, and then select an appropriate route for the video flow.

2 Related work

Network management aimed at improving video QoE has been studied so far [3], [4], [5]. To achieve a network-wide QoE fairness, paper [4] allocated network resource for heterogeneous applications dynamically by using SDN. In order to improve the quality of video streaming and file download, they took buffering time and throughput into consideration. However, since they do not use a common metric for these applications, the method cannot improve the performance of any applications except video and file download. By using QoE as a metric for network management, the same management policy for controlling any applications can be applied even when various applications coexist. Therefore, measuring QoE by using in-network information and performing QoE-based control make network management elastic.

Paper [5] proposed a QoE-based route optimization for multimedia services to maximize end users' QoE. To identify the best route for each flow in SDN controller, they made Session Initiation Protocol (SIP) server mediate between client and media server, and obtained media parameters, such as video codec, from SIP server. However, all multimedia services do not use intermediate node such as SIP server in real Internet. Therefore, in this paper, we consider the way of predicting not only network parameters but also media parameters based on available in-network information.

3 QoE calculation model

We explain QoE calculation for two applications, that is, not only for video QoE our method focuses but also for file transfer, because any kind of applications coexist in the real Internet.

3.1 QoE calculation for video streaming services

QoE calculation for video streaming services is standardized in ITU-T G.1071 [6]. G.1071 requires network condition measured at end hosts, and video settings. Although QoE value is calculated in the combination of video and audio metric, we here focus only on the video in this study because it is a primary factor of video QoE. Note QoE is ranged from 1 to 5.

G.1071 consists of two resolution categories: high resolution and low resolution (Table 1). We here use high resolution because high resolution video is more general than low one. Those equations are as follows:

Table 1 The target video settings in G.1071

Category	Lower resolution	Higher resolution
Protocol	RTSP over RTP	MPEG2-TS over RTP
Video codec	H.264，MPEG-4	H.264
Resolution	QCIF(176×114), HVGA(480×320)	SD(720×480), HD(1280×720，1920×1080)
Video bitrate(bps)	QCIF:32～1000 k, HVGA:192～6000 k	SD:0.5～9 M, HD:0.5～30 M
Video framerate	5～30 fps	25～60 fps

$$\mathrm{QoE}_{\mathrm{video}} = 1.05 + 0.385 \times Q_V + Q_V(Q_V - 60)(100 - Q_V) \times 7.0 \times 10^{-6}, \quad (1)$$

$$Q_V = 100 - Q_{codV} - Q_{traV}, \quad (2)$$

$$Q_{codV} = A \times e^{B \times b_p} + C + (D \times e^{E \times b_p} + F) + G, \quad (3)$$

$$b_p = \frac{b_r \times 10^6}{r \times f_r}, \quad (4)$$

$$Q_{traV} = H \times \log(I \times p_{lc} + 1), \quad (5)$$

$$p_{lc} = J \times \exp[K \times (L - M) \times \frac{p_{lr}}{M \times (N \times p_{lb} + O) + p_{lr}}] - J. \quad (6)$$

The range of Q_V is from 1 to 100, and Q_V is converted to the QoE value by the equation (1). Parameters of $A \sim O$ are fixed and defined in G.1071, and take positive except B and E. Besides, parameters of video bitrate b_r [bps], resolution r [pixel], frame rate f_r [fps], and packet loss concealment (PLC) are pre-determined as the video settings. On the other hand, parameters of packet loss rate p_{lr} [%] and average number of consecutive packet losses p_{lb} need to be measured at end hosts. The values of fixed parameters in the equation (6) (i.e., $J \sim O$) have different values in accordance with PLC which is an application function to correct a damaged video frame happened at packet losses. PLC consists of Freezing, which only ignores losses, and Slicing, which tries to correct the losses.

3.2 QoE calculation for file transfer

As paper [7] proposed QoE calculation for file transfer application (FT), we here explain how to calculate QoE.

$$\mathrm{QoE}_{\mathrm{FT}} = \begin{cases} 1 & (R \leq R^-) \\ a \cdot \log_{10}(b \cdot R) & (R^- < R < R^+) \\ 4.5 & (R^+ \leq R). \end{cases} \quad (7)$$

$\mathrm{QoE}_{\mathrm{FT}}$ is ranged from 1 to 4.5. R is the goodput of flow. R^+ is the maximum transmission speed in network without packet losses. In this study, we use 100 Mbps, which is the maximum transmission speed in our experimental network, as R^+. On the other hand, as the lowest bandwidth provided by AT&T's DSL service is 0.8 Mbps [8], we set the minimum transmission speed, which are referred to as R^-, to 0.8 Mbps. a and b are determined by fitting the approximate curve in the equation (7). As a result, a and b are calculated as following: $a = 1.67$, $b = 4.97$.

4 SDN-based early QoE prediction and route selection

In this section, we propose an early QoE prediction method and a route selection method. To predict video QoE for arrival flow, we transmit a new flow on a temporary route for very short period and calculate QoE from the network information measured during that period. After the QoE prediction, we switch the flow to an appropriate route based on the predicted QoE.

4.1 Early QoE prediction at the flow arrival timing

As explained in section 3.1, 6 parameters are required for QoE calculation of video flow. However, the average number of consecutive packet losses and PLC cannot be obtained in a network because OpenFlow cannot count the consecutive number of packet losses and PLC is an application parameter invisible on network. Thus, in this study, we set these parameters to fixed value, considering the case bringing the worst QoE, as does the previous study [2]. Although other 4 parameters have to be obtained, obtaining the precise information is extremely difficult at the time of flow arrival. Therefore, we predict these parameters by combining the information of network condition collected before the flow arrival and short-period measurement after the flow arrival. Our QoE prediction method consists of two functions: (1) packet loss rate prediction and (2) video settings prediction.

4.1.1 Packet loss rate prediction

As packet losses are basically measured from flows which have been transmitting already, it is essentially difficult to obtain it before start of new flow transmission. However, we can assume the packet loss ratio (PLR) is the same for any flows on a same network. Hence, we measure the PLR on existing (ongoing) traffic regularly and use the value as the PLR for QoE prediction of new arrival flow. Here, we have one more assumption in which the arrival flow will not suffer from overload if it is forwarded on the network.

To measure the PLR of existing traffic, we use the statistic information, called PortStats, of OpenFlow. PortStats can be collected from OpenFlow Switches (OFSs) by OpenFlow Controller (OFC). Since PortStats includes the number of transmitted packets and received packets on each network interface (not each flow), the OFC can calculate the PLR of existing traffic by the difference of the number of transmitted and received packets between neighboring two OFSs. However, PLR may contain measurement errors because PortStats cannot be collected at the exact same time between two OFSs. We handle this measurement error at the same way with our previous study [2]. Briefly explaining, if the number of received packets at the receiver-side OFS is larger than the transmitted packets at the sender-side OFS, we

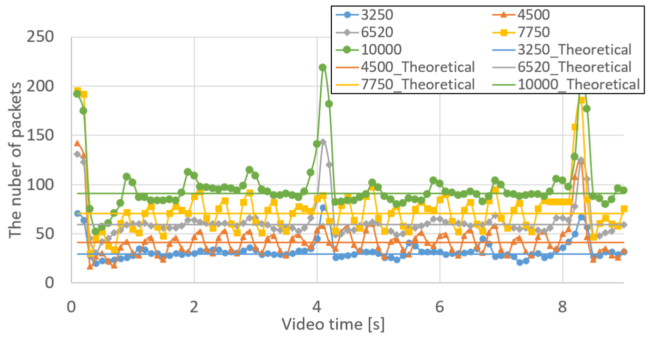

Fig. 1 The number of video flow's transmitted packets every 0.1 sec.

can clearly treat them as measurement errors. We then hold the difference as the accumulated number of surplus packets for correcting subsequent errors. After that, when the number of transmitted packets at the sender-side is larger than the received packets at the receiver-side, we can expect that the difference is the measurement error if we have remaining surplus packets at that time. In such case, we subtract the accumulated number of surplus packets from the difference and then treat the result as the number of packet losses. To conduct the PLR measurement with error correction, we collect PortStats every 1 second in this study.

4.1.2 Video settings prediction

Although it is possible to identify video bitrate based on the measured throughput [2], we cannot measure the throughput before the start of flow transmission. In order to get video bitrate, we need to temporarily forward a new flow on a route with the largest residual bandwidth, which has less possibility for packet losses, for very short period, and measure the number of packets or bytes during the transmission on that route.

Figure 1 shows the number of packets measured for every 0.1 second after the flow transmission starts. Note we only transmit a single video flow on a stable network in this experiment. We vary the video bitrate from 3,250 kbps to 10,000 kbps. Besides, "Theoretical" means the theoretical number of packets that can be transmitted for 0.1 second. Since the video buffering function works at the beginning, the number of packets is significantly different from the theoretical value, irrespective of video bitrate. This causes the overestimation of video bitrate, and thus it is hard to use that value for identifying video bitrate.

Due to this throughput fluctuation, QoE measurement method for on-going flows proposed in our previous study used the measurement results for 8 seconds from first second to ninth seconds. That is why we have to complete the QoE prediction and the initial route selection until 1 second so as to enable the QoE measurement on a particular route. As there are delays such as transmission and processing of PortStats, we use the number of bytes measured for a little bit less than 1 second, i.e., 0.9 seconds, after the flow arrival. For this, we investigate the characteristics of

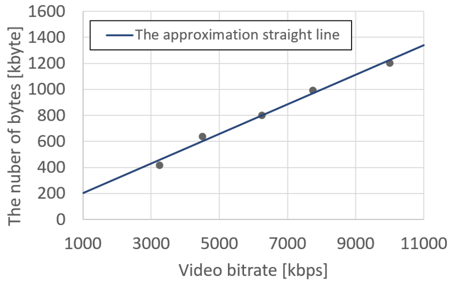

Fig. 2 The number of transmitted bytes for 0.9 seconds after a flow starts.

video packet transmission for 0.9 seconds immediately after the flow transmission starts. Figure 2 shows the number of transmitted bytes for 0.9 seconds, when the video bit rate is varied. In Figure 2, we can expect that the number of bytes is proportional to the video bitrate even involving the effect of video buffering. Based on this assumption, we perform straight linear approximation as following:

$$R_v = 8.7903 \times B_v - 7.7067 \times 10^2 \tag{8}$$

where R_v [kbps] is video bitrate and B_v [kbyte] is the number of bytes for 0.9 seconds from the start of new video flow. As a result, we predict video bitrate from the approximation straight line (equation (8)).

In OpenFlow, OFC can counts B_v by exploiting *packet-in* message, which is used by OFS for asking a way to handle the new flow for the OFC. Specifically, when a new flow arrives at a OFS, the OFS transmits a *packet-in* message with the header of that packet to the OFC. The OFC generally returns a flow control rule, called *flow entry*, as a *flow mod* message and then the OFS will not send *packet-in* after receiving *flow mod* message. However, in this study, we make the OFC return *packet-out* message only, which instructs to send the particular packet to next hop. After 0.9 seconds pass, the OFC returns a *flow mod* that matches every packets of the flow. In this way, an OFS sends *packet-in* whenever new packets arrive until the 0.9 seconds, and the OFC counts the total bytes transmitted during this period. Note that, if there are multiple OFSs on the selected route, we conduct this process only for the first OFS (nearest to the sender) and send a *flow mod* controlling every packets of the flow to the other OFSs. We can predict video bitrate based on the equation (8) by using B_v obtained in this way.

Frame rate and resolution are predicted based on this predicted video bitrate. We use Table 2 which is made from the recommendation of video settings of YouTube in terms of resolution, frame rate, and video bitrate [9] as in the previous study.

4.2 Route selection based on the predicted QoE

After video settings prediction, the OFC calculates QoE based on both the PLR of existing traffic and predicted video settings. Here, as a network condition may be

Table 2 Estimation of resolution and frame rate based on video bitrate.

Video bitrate [kbps]	Resolution	Frame rate [fps]
∼3,250	SD(720×480)	30
3,250∼4,500	SD(720×480)	60
4,500∼6,250	HD(1280×720)	30
6,250∼7,750	HD(1280×720)	60
7,750∼10,000	HD(1920×1080)	30
10,000∼	HD(1920×1080)	60

different among routes, the OFC measures the PLR of existing traffic and predicts QoE for each route. After the QoEs of all available routes are predicted, the OFC tries to choose an appropriate route for the new arrival flow for maximizing QoE of the flow. Specifically, we select a route which has the highest predicted QoE. By doing that, we can forward a video flow to a route with having the largest QoE by 1 second after the new flow arrival.

5 Experimental evaluation

We conduct experiments to evaluate the proposed method. The goal of this experiment is to show effectiveness of the QoE prediction. At first, we compare predicted QoE with actual QoE that can be obtained at end hosts. Then, we show that the QoE-based route selection method successfully improves QoE of all flows transmitted on the SDN-enabled network.

5.1 Comparative method

We use a throughput-oriented route selection [1] as the conventional method. Briefly explaining, to avoid packet losses for a new arrival flow, the method temporarily forwards the flow on a route with the largest residual bandwidth because its required throughput is unknown at that time. After the OFC measures the throughput based on the FlowStats which is the statistic information of each flow, the OFC forwards the flow to the route with the smallest but sufficient residual bandwidth. By doing this throughput-oriented mechanism, since a route with the largest residual bandwidth is prepared for next flow, we can avoid packet losses at the next flow arrival as much as possible.

5.2 Experimental settings

The experimental topology is shown in Figure 3. Regarding OpenFlow, we use Trema as OFC and install OpenvSwitch on every OFSs. All devices are connected with 100 Mbps Ethernet. We put a Linux PC between OFS 1-2 and OFS 2-2 to generate 1 % random packet loss. We define the route from OFS 1-1 to OFS 2-1 as Route 1 and from OFS 1-2 to OFS 2-2 as Route 2. In addition, we prepare two back-

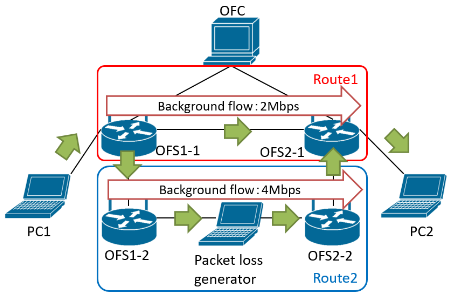

Fig. 3 Experimental environment.

Table 3 The true value of QoE after the route selection in scenario 1.

Video	The proposed method			The conventional method		
bitrate	Median	Max.	Min.	Median	Max.	Min.
3,250 kbps	4.395	4.395	1.317	1.311	1.328	1.285
10,000 kbps	4.391	4.391	1.302	1.321	1.345	1.307

ground traffic flows, which are a 2 Mbps flow through Route 1 and a 4 Mbps flow through Route 2 to make an imbalanced residual bandwidth between two routes. After that, we evaluate how the QoEs of all flows are varied when several flows including video streaming are transmitted from PC 1.

5.3 *Evaluation of the early QoE prediction*

To show effectiveness of our proposed QoE prediction method, we conduct two scenarios with different evaluation purposes.

- Scenario 1: We show how the proposed method improves the QoE of the incoming video streaming flow. We also evaluate accuracy of predicted PLR, video bitrate and QoE.
- Scenario 2: We show how the proposed method is effective by using predicted QoE for the route selection in case of multiple new flow arrivals.

5.3.1 Scenario 1: Evaluation in case of only video flow

In this scenario, we show the performance of the route selection. Specifically, we transmit a video flow from PC 1 to PC 2 and evaluate the result of route selection. In this experiment, we transmit two kinds of video, that are 3,250 kbps and 10,000 kbps and do this 9 rounds.

Table 3 shows the true value of QoE after the route selection. In the conventional method, although a new flow is transmitted on Route 1 which has the largest residual bandwidth at first, the video flow is finally switched to Route 2 due to the throughput-oriented mechanism. As a result, since packet losses occur in Route 2,

Table 4 The predicted and true vlue of PLR in Route 1.

Video bitrate	Predicted value Median	Max.	Min.	True value
3,250 kbps	0 %	1.622 %	0 %	0 %
10,000 kbps	0 %	3.125 %	0 %	0 %

Table 5 The predicted and true vlue of video bitrate in Route 1.

Video bitrate	Predicted value Median	Max.	Min.	True value
3,250 kbps	3,128 kbps	3,317 kbps	3,057 kbps	3,500 kbps
10,000 kbps	5,339 kbps	5,872 kbps	4,086 kbps	10,000 kbps

Table 6 The predicted and true vlue of QoE in Route 1.

Video bitrate	Predicted value Median	Max.	Min.	True value
3,250 kbps	4.720	4.726	1.286	4.395
10,000 kbps	4.732	4.747	1.393	4.395

QoE of the video flow drastically drops to around 1.3. On the other hand, in the proposed method, the video flow is continuously transmitted on Route 1 as a result of the route selection based on predicted QoE. Thus, the median and maximum value of QoE are around 4.3, which is good quality. However, the minimum value of QoE is around 1.3. This is because the occurrence of measurement errors of PLR, which cannot be corrected the way described in Section 4.1.1, degrades the predicted QoE, thereby accordingly switching the video flows to the Route 2.

To investigate the measurement error, Tables 4, 5 and 6 show the predicted and true value of PLR, video bitrate and QoE in Route 1, respectively. Although no packet loss happens in Route 1 in fact, the maximum value of predicted PLR indicates more than 1% larger than the true value. The gaps are caused by miss-correction of errors introduced in Section 4.1.1. Specifically, measurement errors cannot be corrected because of a mistake in judgement of packet loss and measurement error. Thus, the predicted QoE is degraded by this measurement error.

Next, focusing on video bitrate, the predicted video bitrate of 3,250 kbps is almost same with the true value, while 10,000 kbps is lower than the true value at all. This is because there is limitation on the number of bytes measured in OFC due to its processing delay. However, the difference between the true value and median/max. predicted value is only around 0.3, that is almost same with the true value because video bitrate has the less impact on QoE [2]. From these results, we can demonstrate that the proposed method can improve video QoE by switching to the

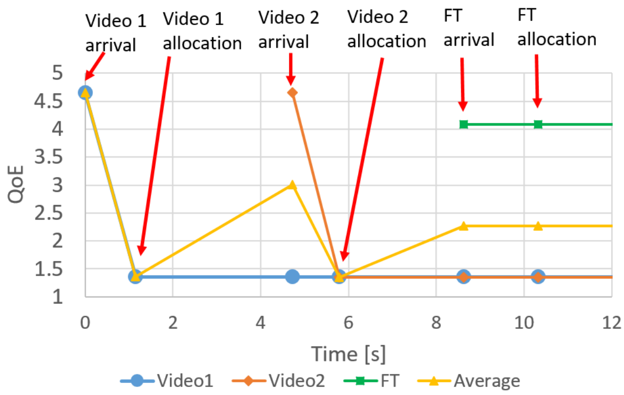

Fig. 4 QoE transition of the conventional method in scenario 2.

route based on predicted QoE. In addition, since the predicted QoE value shows the almost same value independent of the change in the video bitrates, we can say that the proposed method has video bitrate tolerance characteristics.

5.3.2 Scenario 2: Evaluation in case of multiple flows

In this scenario, we demonstrate the effectiveness of the proposed method, using predicted QoE for the route selection, in case of multiple new flow arrivals. We transmit three flows in the order of video flow 1, video flow 2, and File transfer (FT) flow every 5 seconds. The video bitrate of both video 1 and video 2 is set to 2,500 kbps and the FT flow is a simple TCP file transfer. Here, we assume that OFC can precisely predict QoE of FT, and use the true value of QoE for route selection of FT. We conduct this experiment at 9 times, but here show the result of the median value.

Figure 4 shows true value of QoE in the conventional method in the time series. Although a new flow is temporarily transmitted on Route 1 which has the largest residual bandwidth, that video flow is then switched to Route 2 based on the throughput-oriented mechanism. As a result, QoE of video 1 is high at the flow arrival but its QoE drops drastically after the route selection due to packet losses on Route 2. For the video 2, as the conventional method performs the same flow management as with video 1, QoE of video 2 drops. Lastly, for the FT, the conventional method temporarily selects Route 1, but does not change the route because the residual bandwidth of Route 2 is less than that of Route 1. As a result, FT can maintain high QoE but both videos cannot in the conventional method.

Figure 5 shows true value of QoE in the proposed method in the time series. The video 1 flow is temporarily transmitted on Route 1 as with the conventional method, but decides to keep the route on the same one based on the QoE prediction. The video 2 is temporarily transmitted on Route 2 at arrival timing because Route 2 has the largest residual bandwidth at that time. Thus, QoE of video 2 is low at the flow arrival. However, since the proposed method switches the video 2 flow to Route 1 in accordance with its predicted QoE, QoE of the video 2 flow is clearly

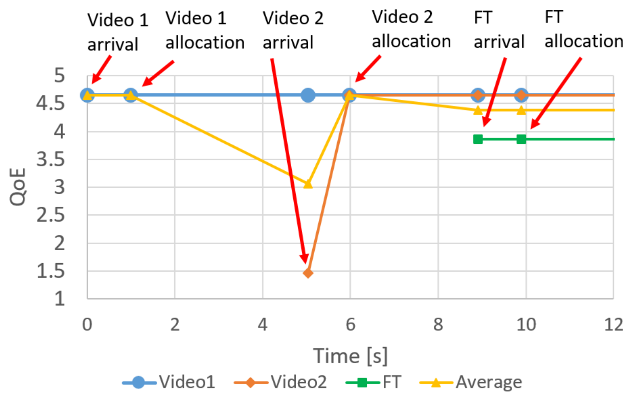

Fig. 5 QoE transition of the proposed method in scenario2.

improved after 1 second. Finally, FT is first temporarily transmitted on Route 2, which has the largest residual bandwidth, and then does not change the transmission route. Hence, QoE of FT is kept around 3.5 because QoE of FT strongly depends on throughput performance. As a result, the proposed method can keep QoE of every flows excellent level. Therefore, we can remark that the predicted QoE-based route selection successfully improves QoE for multiple new flow arrivals.

However, the proposed method still has the following limitations: it does not consider that QoE of existing flows may drop due to transmission of a new flow. If the route selection for a new flow causes the exceed of available bandwidth, QoE of existing video flows inherently drops. Moreover, since the proposed method does not prioritize to prepare a route with the highest residual bandwidth, the possibility of bandwidth scarcity at flow arrivals is relatively higher than that of the conventional method.

6 Conclusion

As G.1071 requires the network measurement of at least 8 seconds to calculate QoE, a QoE-based route selection is quite difficult at the arrival timing of new flow. To resolve this problem, we proposed a early QoE prediction method for new arrival video flow. As both PLR and video settings are required for the QoE calculation, we predicted PLR by exploiting existing traffic, and video settings by exploiting the measurement result of very-short period (0.9 seconds) immediately after the video start while alleviating the effects of video buffering. The proposed method then selected an appropriate route for every flows in accordance with the predicted QoE. In our experiments, we showed that the proposed method can predict QoE precisely irrespective of the change in the video bitrate. We also demonstrated that the route selection for arrival video and file transfer flows based on predicted QoE successfully improved their QoE. As a next step, we are going to collaborate the QoE prediction method, which initially selected an appropriate route, and our proposed QoE estimation method for ongoing flows [2] in order to catch up with the change of network conditions in the timely manner.

Acknowledgements This paper was partly supported by JSPS KAKENHI Grant Number 18H03234, NICT Grant Number 19304, USA Grant number NSF 1818884 and 1827923.

References

1. M. Tagawa, Y. Taenaka, K. Tsukamoto, and S. Yamaguchi, "A Channel Utilization Method for Flow Admission Control with Maximum Network Capacity toward Loss-free Software Defined WMNs," the Fourteenth International Conference on Networks (ICN 2015), pp. 118-123, Febrary 2016.
2. S. Shimokawa, T. Kanaoka, Y. Taenaka, K. Tsukamoto, and M. Lee, "SDN-based time-domain error correction for in-network video QoE estimation in wireless networks," the 11th International Conference on Intelligent Networking and Collaborative Systems (INCoS 2019), pp 331-341, September 2019.
3. A. Erfanian, F. Tashtarian and M. H. Yaghmaee, "On Maximizing QoE in AVC-Based HTTP Adaptive Streaming: An SDN Approach," 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), pp. 1-10, June 2018.
4. T. Zinner, M. Jarschel, A. Blenk, F. Wamser and W. Kellerer, "Dynamic application-aware resource management using Software-Defined Networking: Implementation prospects and challenges," 2014 IEEE Network Operations and Management Symposium (NOMS), pp. 1-6, May 2014.
5. O. Dobrijevic, A. Kassler, L. Skorin-Kapov, and M. Matijasevic, "Q-POINT: QoE-driven path optimization model for multimedia services," Wired/Wireless Internet Communications (WWIC 2014), pp. 134147, May 2014.
6. Recommendation ITU-T G.1071, "Opinion model for network planning of video and audio streaming applications," November 2016.
7. S. Thakolsri, S. Khan, E. Steinbach, and W. Kellerer, "QoE-driven cross-layer optimization for high speed downlink packet access," Journal of Communications 4 (9), 669-680, October 2009.
8. AT&T Speed Tiers, https://att.net/speedtiers.
9. Recommended upload encoding settings, YouTube Help, https://support.google.com/youtube/answer/1722171?hl=en.