

Customized Transmission Protocol for Tile-Based 360° VR Video Streaming Over Core Network Slices

Yannan Wei^{ID}, *Graduate Student Member, IEEE*, Qiang Ye^{ID}, *Senior Member, IEEE*, Kaige Qu^{ID}, *Member, IEEE*, Weihua Zhuang^{ID}, *Fellow, IEEE*, and Xuemin Shen^{ID}, *Fellow, IEEE*

Abstract—Tile-based streaming has been proposed to address the challenge of high transmission rate demand in 360° virtual reality (VR) video streaming. However, it suffers from network and viewing behavior dynamics (i.e., head movements), while encoded video tiles have various properties in terms of transmission priority, deadline, and reliability requirement. Hence, a supporting transmission protocol is imperative. In this paper, we propose a customized transmission protocol based on Quick UDP Internet Connections (QUIC) which operates over a VR video network slice in the core network. The QUIC protocol is tailored to accommodate the characteristics of tile-based VR video streaming where explicit mapping relations between requested video tiles and QUIC streams are established. Two customized in-network protocol functionalities including packet filtering and caching-based packet retransmission are proposed, to filter out outdated video data due to field-of-view (FoV) prediction errors under viewing behavior dynamics and to achieve efficient packet retransmissions with disparate transmission reliability requirements. A slice-level packet header is designed to support enhanced slice-based VR video transmission with the proposed protocol functionalities. Key transport parameters are determined via theoretical analysis. Simulation results are presented to demonstrate the effectiveness of our proposed transmission protocol in achieving short average video segment downloading time and high average video segment quality.

Index Terms—360° VR video streaming, transmission protocol, QUIC, tile-to-stream mappings, FoV, protocol functionalities, SDN/NFV-enabled network slice.

I. INTRODUCTION

360° virtual reality (VR) video streaming has attracted significant attention from both academia and industry due to immersive user experience and enormous vertical markets such as entertainment [2] and education [3]. It requires extremely high transmission rate (e.g., 25 Mbps) and low

Received 2 June 2024; revised 16 September 2024; accepted 11 October 2024; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor Y. Chen. This work was supported by the research grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada. This work was presented in part at IEEE/CIC ICCC 2024 [DOI: 10.1109/ICCC62479.2024.10681889]. (Corresponding author: Kaige Qu.)

Yannan Wei, Kaige Qu, Weihua Zhuang, and Xuemin Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: y272wei@uwaterloo.ca; k2qu@uwaterloo.ca; wzhuang@uwaterloo.ca; sshen@uwaterloo.ca).

Qiang Ye is with the Department of Electrical and Software Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: qiang.ye@ucalgary.ca).

Digital Object Identifier 10.1109/TNET.2024.3485583

latency for smooth high resolution (e.g., 4K) spherical video playback [4]. Typically, a viewer (or video client) wears a head-mounted display (HMD) to watch a panoramic VR video. At any time instant, the viewer watches only part of the video due to the limited span of the HMD (e.g., 120° × 120°), referred to as field-of-view (FoV), with the viewpoint in the center. To address the challenge of high transmission rate demand, tile-based adaptive 360° VR video streaming has been proposed. Specifically, at the server side, a spherical VR video representation is first transformed into a planer format using techniques such as the equirectangular projection (ERP) [5]. Then, the projected VR video is temporally divided into a sequence of video segments, each of which is further spatially partitioned into multiple non-overlapping video tiles. At the viewer side, head movements are tracked for FoV prediction [6]. Based on the FoV prediction results and the estimated transmission rate, a viewer selectively requests video tiles with different bitrates to enhance the viewing experience [7], [8], [9].

Encoded video tiles have various properties in terms of transmission priority, deadline, and reliability requirement. First, with the scalable high-efficiency video coding (SHVC) [10], each video tile can be encoded into one base layer (BL) and multiple enhancement layers (ELs). BL tiles ensure video smoothness and have a higher transmission priority than EL tiles that enhance video quality. Second, each video tile has a strict deadline to be delivered. Especially, if viewers suddenly rotate heads, additional urgent video tiles should be requested with a small deadline, referred to as the motion-to-photon (MTP) latency requirement (usually less than 20 ms [11]), to compensate the current viewing experience. Third, BL packets have a higher transmission reliability requirement than EL packets. Deadline-violated EL packets can be directly discarded without retransmissions for congestion alleviation and high transmission efficiency [12], while BL packets should always be reliably transmitted. In addition, tile-based VR video streaming suffers from both transmission rate variations and viewing behavior dynamics which are constantly driven by head movements and cause FoV prediction errors. Therefore, to support smooth and high-quality VR video streaming, a customized transmission protocol which accommodates various video tile properties and promptly reacts to network and viewing behavior dynamics should be developed.

Driven by the software-defined networking (SDN) and network function virtualization (NFV) technologies, multiple virtual networks, also known as *network slices*, can be created over a shared physical network for supporting diversified services with different performance requirements [13], [14]. A network slice with flexible resource orchestration can be deployed to support the 360° VR video streaming service for finer-grained quality-of-service (QoS) guarantee, where optimal routing path(s) can be established with dedicated processing and transmission resources reserved on NFV nodes and physical links along the route. Customized protocol functionalities and operations can be embedded on network nodes for flexible and efficient transmission control.

We consider an SDN/NFV-based end-to-end (E2E) transmission network where a VR video network slice is deployed between each pair of ingress and egress nodes in the core network. The Quick UDP Internet Connections (QUIC) protocol is adopted as the base transport protocol for E2E video transmission. For the tile-based 360° VR video streaming service, several research issues regarding transport protocol design should be investigated. First, various video tile properties should be accommodated in protocol operations. Optional header fields are thus needed in the QUIC packet header to indicate each VR video packet's properties. Video tiles with disparate properties should not be transmitted over the same QUIC stream(s), and only data of video tiles with the same properties should be multiplexed in a QUIC packet. Correspondingly, the mapping relations between requested video tiles and QUIC streams should be determined, and how to conduct streaming multiplexing and QUIC packet assembly needs to be revisited. Second, a slice-level transmission scheme should be designed to support enhanced VR video transmission over a VR video slice, where differentiated packet loss recovery mechanisms are required for BL and EL packets. In addition, when viewers suddenly rotate heads, which causes FoV prediction errors, some requested video tiles for the previously predicted FoV become outdated. Enroute VR video packets may contain outdated video data which should be dropped to save transmission resources.

To address the above research issues, in this paper, we present a customized transmission protocol based on QUIC which operates over a VR video slice in the core network. Specifically, the QUIC protocol is first tailored to accommodate the characteristics of tile-based VR video streaming, where an explicit one-to-one mapping between requested EL tiles and QUIC streams is established, and the behaviors of stream multiplexing and QUIC packet assembly at the server side are modified accordingly. Two customized protocol functionalities are designed, including packet filtering and caching-based packet retransmission. The packet filtering functionality removes outdated video data in prompt response to viewing behavior dynamics (i.e., head movements), and the caching-based packet retransmission functionality performs efficient packet retransmissions according to disparate reliability requirements. A slice-level packet header is designed to support enhanced slice-based VR video packet transmissions with the proposed protocol functionalities. The main contributions of this paper are summarized as follows.

- The QUIC protocol is tailored to better support the considered tile-based 360° VR video streaming service where explicit mapping relations are established between requested video tiles with various properties and QUIC streams;
- Customized protocol functionalities including packet filtering and caching-based packet retransmission are proposed to support responsive and efficient transmission control. A slice-level packet header is designed to support enhanced VR video transmission with the proposed protocol functionalities;
- Key transport parameters in the proposed transmission protocol, such as the minimum required caching-buffer sizes for BL and EL packets, are determined based on analytical modeling.

The remainder of this paper is organized as follows. Section II gives an overview of related work. The system model is presented in Section III. The main designs in developing our customized slice-level transmission protocol and the corresponding E2E VR video transmission process are discussed in Section IV. In Section V, key transport parameters in the proposed transmission protocol are determined via theoretical analysis. Performance evaluation of the proposed transmission protocol is presented in Section VI, and conclusions of this work are drawn in Section VII.

II. RELATED WORK

A. Prioritized Transmission Scheduling of Requested Video Tiles

To enhance the quality-of-experience (QoE) of tile-based 360° VR video streaming, existing studies on protocol design conduct prioritized transmission scheduling among requested video tiles at the client side or at the server side, while considering various video tile properties. For example, at the client side, with the considerations of the importance of video smoothness over video quality and FoV prediction errors, BL tiles are always downloaded first until a preset upper bound of playback buffer occupancy is reached [15]. When a viewer's future FoV is not correctly predicted, new instant EL tiles for the updated FoV are immediately requested before the predictive EL tiles [16]. At the server side, different schedulers are developed to schedule and transmit requested video tiles to a video client. The proposed schedulers determine and update the *scheduling priorities* of the requested video tiles and schedule the transmission of video tiles according to their scheduling priorities. The scheduling priority is usually determined by considering each requested video tile's priority, playback deadline, remaining size, and current network conditions [17], [18], [19]. In addition, the stream multiplexing and stream scheduling features that are naturally supported in the QUIC protocol can be leveraged to achieve prioritized transmissions among requested video tiles [20], [21]. In a QUIC connection, multiple video tiles can be concurrently requested via different QUIC streams, and different scheduling policies can be flexibly designed at the server side to determine how streams are multiplexed and scheduled for different transmission purposes. For example, when viewers suddenly

rotate their heads, QUIC allows the server to quickly send urgent video tiles with a high priority to mitigate the negative impacts due to missing tiles (e.g., motion sickness) [22]. QUIC also provides flexible stream-level data transmission control through sending signaling frames such as RESET_STREAM and STOP_SENDING frames to adapt to viewing behavior dynamics. Most existing works focus on improving protocol operations at end hosts to enhance the VR video transmission performance, which may lead to slow responsiveness to network and viewing behavior dynamics (e.g., random packet loss or head movements). To accommodate various VR video packet properties and promptly react to network and viewing behavior dynamics during packet transmissions, in this paper, we develop a customized transmission protocol based on QUIC which operates over a VR video network slice in the core network with in-network protocol operations for enhanced VR video transmission.

B. SDN/NFV-Based Transmission Protocol Design

In the SDN/NFV framework, flexible and prompt transmission control can be realized with in-network awareness. Specifically, SDN switches, as configured by an SDN controller via southbound protocols (e.g., OpenFlow [23]), can achieve early congestion detection and throttle the sending rate by modifying the *Receive Window* field in a Transmission Control Protocol (TCP) acknowledgement (ACK) packet or by selectively discarding packets of different traffic flows to trigger fast retransmission [24], [25]. With the NFV technology, customized protocol functionalities can be deployed at in-network nodes to realize in-network transmission control for enhancing the E2E transmission performance of a particular service [26], [27]. For example, for delay-sensitive and loss-tolerant 2D video streaming service, a selective caching function is deployed at an ingress node of a 5G core network to adaptively cache EL packets for alleviating network congestion with limited video quality degradation, and the cached EL packets are resumed transmissions when network conditions are better [28]. In this paper, we propose two customized in-network protocol functionalities under the SDN/NFV framework, including packet filtering and caching-based packet retransmission, for supporting enhanced VR video transmission. The packet filtering functionality removes outdated EL video data due to FoV prediction errors in prompt response to viewing behavior dynamics, and the caching-based packet retransmission functionality efficiently retransmits lost BL and EL packets with disparate transmission reliability requirements.

III. SYSTEM MODEL

A. Network Model

An SDN/NFV-based E2E transmission network is considered to support (on-demand) 360° VR video streaming service delivered from remote VR video servers to video clients. As shown in Fig. 1, geographically adjacent video clients connect via different base stations (e.g., gNodeBs) to an edge node (e.g., an egress edge switch of its connected

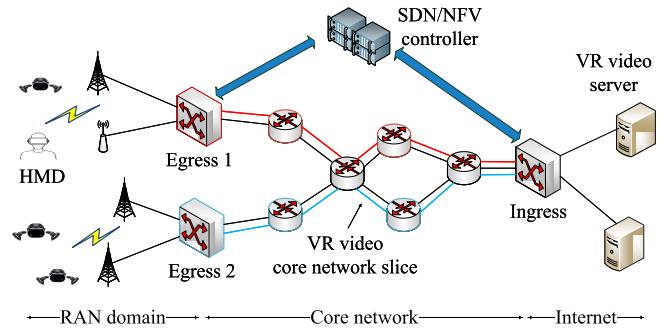


Fig. 1. The SDN/NFV-based E2E transmission network scenario.

core network). Ingress and egress edge switches (referred to as nodes) are located in between the radio access network (RAN) domain and the core network and in between the core network and the Internet, respectively. Ingress and egress nodes host advanced traffic management functions and are deployed by a mobile operator that provides network support for the provisioning of the 360° VR video streaming service. SDN switches including ingress and egress nodes are physically (may not be directly) connected to the SDN/NFV controller. The logical centralized SDN/NFV controller in the control plane has a global view over the underlying core network, and communicates with programmable SDN switches including ingress and egress nodes in the data plane via southbound protocols such as the OpenFlow [23]. Under the SDN/NFV architecture, multiple virtual networks or network slices are instantiated over the shared core network for supporting diversified services. Specifically, a VR video slice is deployed between each pair of ingress and egress nodes in the core network for supporting aggregated VR video traffic. For each VR video slice, a dedicated virtual network topology (assumed linear for simplicity) is configured by the SDN/NFV controller [29], where $W - 1$ intermediate SDN switches $\{s_1, \dots, s_{W-1}\}$ between a pair of ingress (s_0) and egress (s_W) nodes are interconnected by W transmission links, denoted by $\{l_0, \dots, l_{W-1}\}$. Virtual network resources including link transmission resources and processing resources are reserved at physical links and programmable switches along the route for data transmission and processing. Customized protocol functionalities are enabled at network nodes within each slice to achieve finer-grained VR video service provisioning.

E2E VR video transmission traverses three network segments: (i) from a VR video server on the Internet to an ingress node, (ii) over a VR video slice, and (iii) from the egress node of the VR video slice to a video client. Here, we focus on the core network and aim to develop a slice-level transmission protocol for supporting enhanced VR video data transmission which operates over a VR video (core network) slice. We adopt QUIC as our protocol design base for E2E video data transmission due to its superiority in achieving reduced E2E connection establishment latency and in stream multiplexing with flexible stream-level transmission control [30].

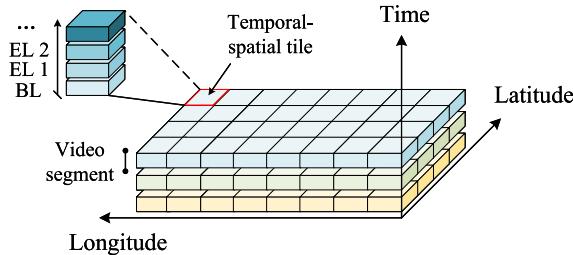


Fig. 2. Tile-based VR video encoding.

B. Video Traffic Model

As shown in Fig. 2, a VR video is temporally divided into a sequence of video segments, each with a playback time of typically 2-10 seconds. Each video segment is spatially partitioned into multiple non-overlapping video tiles. Each temporal-spatial video tile is encoded into multiple interdependent layers for adaptive streaming under network transmission rate variations. Specifically, each video tile is encoded into one BL and multiple ELs. BL tiles ensure video playback smoothness and provide basic video quality. Each BL packet is required to be delivered with high reliability, and BL packet losses result in video rebuffering. EL video tiles provide improved video quality and can be decoded only if the corresponding BL tiles are decoded. EL packets can be discarded with low reliability for congestion alleviation, without affecting the video playback smoothness. Therefore, BL packets have a higher transmission priority and reliability requirement than EL packets. The scalable and layered video tile encoding provides additional flexibility in response to network and viewing behavior dynamics in small timescales due to the disparate transmission reliability requirements of BL and EL packets [28].

Suppose that encoded VR videos are stored on remote servers on the Internet. Video clients progressively download video content from remote servers on a video segment basis. For each segment downloading, an HMD selectively requests a set of video tiles with different numbers of encoding layers, based on factors such as the estimated throughput, tracked head/viewpoint movement trajectory, and playback buffer occupancy, etc. An HMD sends additional urgent requests after sudden head movements during watching a video segment or when FoV prediction errors are identified. Here, we consider that each VR video server streams full-view BL for video robustness and EL tiles covering a client's FoV.

C. Main Protocol Functionalities

To support enhanced VR video transmission by accommodating various VR video packet properties and by adapting to viewing behavior dynamics, two customized protocol functionalities are proposed.

1) Packet Filtering: When an FoV prediction error occurs due to sudden head movements, previously requested video content may deviate from the updated predicted FoV. In this case, additional urgent EL tiles should be requested. As enroute VR video packets may contain outdated EL video data for streaming the previously predicted FoV, a packet filtering

function is designed to promptly remove outdated video data that is no longer needed by clients, which saves transmission resources while adapting to viewing behavior dynamics. Given the deployment and time costs for packet filtering operations, we consider that the packet filtering function is enabled at only the ingress and egress nodes of a VR video slice. The ingress and egress nodes are equipped with the capability of higher-layer protocol header parsing to identify video data packets containing outdated video data.

2) Caching-Based Packet Retransmission: We consider that the ingress node has a caching buffer for aggregated VR video traffic of each VR video slice. The ingress node temporarily stores copies of VR video packets sent but not acknowledged in the caching buffer for possible retransmissions. The egress node detects any random BL/EL packet loss occurring in the core network and triggers retransmission from the ingress node by using the cached packet copy in the caching buffer, instead of from a remote server to achieve lower packet retransmission delay. In addition, due to the disparate transmission reliability requirements, deadline-violated BL packets are not dropped, and lost BL packets are retransmitted to avoid video rebuffering. Deadline-violated EL packets should be directly dropped without retransmissions. Therefore, separate packet loss detection and retransmission mechanisms are designed for BL and EL packets, respectively, where the capability of differentiating EL packet losses due to deadline violations from those due to random link failures is realized at the egress node.

In addition, due to the limited capacity, caching buffer release is necessary to avoid buffer overflow. We consider two ways of releasing the cached packet copies in the caching buffer of the ingress node: event-driven and time-based. Specifically, caching buffer release can be triggered upon receiving a retransmission request from the ingress node (i.e., event-driven). If no retransmission request reception from the egress node for a certain time, a timeout timer is set at the ingress node to periodically release the cached packet copies in the caching buffer (i.e., time-based). The detailed designs and workflows of the proposed two protocol functionalities are discussed in the next section.

IV. PROTOCOL CUSTOMIZATION FOR VR VIDEO TRANSMISSION

In this section, we elaborate on our main designs in developing the customized slice-level transmission protocol for supporting enhanced VR video data transmission and describe the E2E VR video transmission process with our proposed protocol.

A. Tailored QUIC

To support finer-grained QoS for 360° VR video streaming by accommodating various VR video packet properties (i.e., transmission priority, deadline, and reliability requirement) and adapting to viewer viewing behavior dynamics, we first revisit the QUIC protocol and tailor it from the following three aspects:

- Mapping relations between streams in a QUIC connection and requested video tiles by a client (or an HMD) are

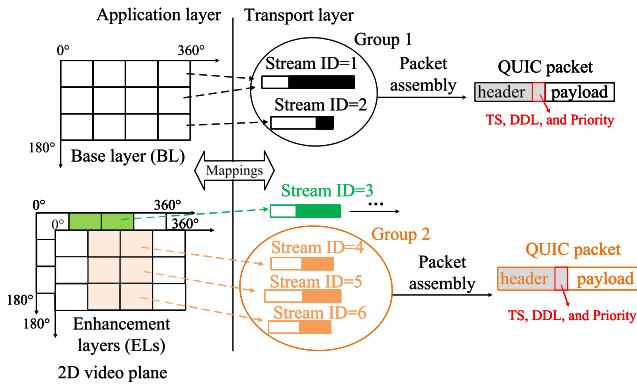


Fig. 3. Tailored QUIC with tile-to-stream mappings.

established. As shown in Fig. 3, a one-to-one mapping between each EL tile and each QUIC stream is established, and a separate set of streams is used to transmit full-view BL. This is achieved by using one stream to carry only one EL tile request and using different streams to carry BL tile requests. Assume the mapping relations between streams and video tiles are known and kept at both the HMD and the server communicating with each other. With the established tile-to-stream mappings, operations to a specific video tile are realized by controlling the data transmission over its corresponding stream, which is helpful for adapting to viewing behavior dynamics caused by head movements.

- Multiple stream groups, each of which consists of the streams corresponding to the requested video tiles of the same (BL/EL) layer, are formed for QUIC packet assembly. One stream group is selected each time to assemble a QUIC packet where round-robin is conducted among the streams within the same group for filling each STREAM frame in a QUIC packet. The approach proposed in [18] can be applied to decide which stream group is selected to assemble a QUIC packet where a *block* in [18] represents a stream group in our case. By doing so, STREAM frames (or video data) assembled in a QUIC packet come from the requested video tiles with the same properties. Video data of different encoding layers is assembled into different types of video packets, including regular BL/EL packets and urgent EL packets; thus, the same set of operations can be enforced on each type of packets.
- Three optional fields, *Timestamp (TS)*, *Deadline (DDL)*, and *Priority*, are added to the QUIC packet header to indicate each packet's transmission priority (i.e., a BL or an EL packet) and deadline [31]. Specifically, the TS and DDL fields are used to record the time when a packet is generated and its transmission deadline, respectively. For uplink transmission, the Priority field of each video request packet is set to 1 for requesting regular video tiles after a video segment downloading is finished and is set to 2 for requesting urgent EL tiles when sudden head movements and/or FoV prediction errors occur. For

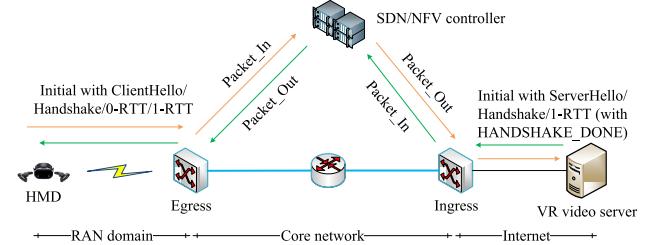


Fig. 4. SDN/NFV-based E2E QUIC connection establishment.

downlink transmission, the Priority field of a BL packet is set to 1, and for an EL packet from the $(n - 1)$ th EL, its priority is set to n ($n > 1$).

B. Connection Establishment

Before video data transmission begins, an E2E QUIC connection needs to be established between an HMD and a VR video server. The SDN/NFV controller with a global view over the physical (core) network can check path availability for two-way communications between a pair of ingress and egress nodes. It configures a network slice and activates customized protocol functionalities (e.g., packet filtering) during the connection establishment phase to support VR video transmission over a VR video slice.

Therefore, different from the typical connection establishment process specified in [30] where signaling messages and data packets (if any) are exchanged directly between an end-device and a server in the data plane, E2E QUIC connection between an HMD and a VR video server in our case is established with the SDN/NFV controller assistance. Specifically, as shown in Fig. 4, the signaling and data packets (if any) exchanged between the HMD and the VR video server during the connection establishment phase, such as the Initial packet with ClientHello/ServerHello, are intercepted by the ingress and egress nodes and are encapsulated into OpenFlow packets (e.g., Packet_In or Packet_Out). These encapsulated packets go through the SDN/NFV controller in the control plane for processing.

After an E2E connection between an HMD and a VR video server is established, the ingress node acts on behalf of the HMD and communicates with the VR video server for packet transmissions on the Internet. The ingress node detects any packet loss on the Internet and triggers the retransmission following any state-of-the-art mechanism proposed based on QUIC [30]. Similarly, the egress node acts on behalf of the VR video server and interacts with the HMD for packet transmissions in the RAN domain. The egress node retransmits any lost packet that happens in the RAN domain. The ingress node interacts with the egress node to regulate the packet transmissions over a VR video slice in the core network. The egress node detects any packet loss and triggers the retransmission from the ingress node.

C. VR Video Data Transmission

- 1) *VR Video Content Request:* As illustrated earlier, each HMD requests video content on a segment basis. Specifically,

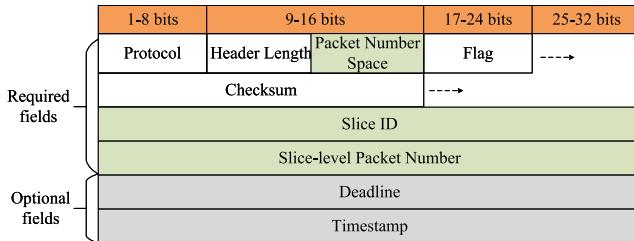


Fig. 5. The slice-level packet header format.

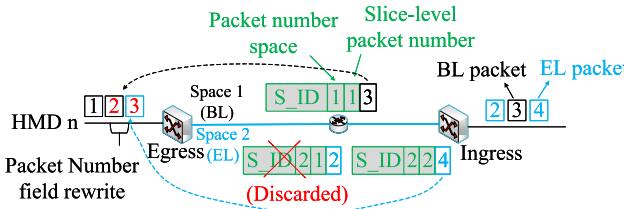


Fig. 6. Video packet transmission over a VR video slice.

it concurrently requests a subset of video tiles of different encoding layers for a video segment. All video tile requests for one segment are multiplexed via different streams in a QUIC connection, as described in Subsection IV-A. Besides, the HMD sends additional requests for urgent EL tiles corresponding to the updated FoV after sudden head movements and/or when FoV prediction errors occur.

2) *Header Conversion at Edge Switches:* A slice-level packet header is designed to support the slice-level VR video packet transmission and the customized protocol functionalities presented in Subsection III-C. To be compatible in terms of protocol operations with the tailored QUIC implemented at end hosts, header conversion and reversion are performed at the ingress/egress node, such as through the tunneling technique [32]. A slice-level packet header format is given in Fig. 5 which includes three important fields, i.e., *Slice ID*, *Packet Number Space*, and *Slice-level Packet Number*. The *Slice ID* field is used for slice identification and packet forwarding, upon which data packets are transmitted along the pre-configured routing path of the indicated core network slice. The fields of *Packet Number Space* and *Slice-level Packet Number* are designated to support the proposed customized protocol functionalities. Specifically, the *Packet Number Space* field is introduced where two separate packet number spaces are used to differentiate between BL and EL packets with disparate transmission reliability requirements. The *Packet Number Space* field achieves logical isolation and thus allows for respective operations (e.g., packet retransmission) to BL/EL packets. BL/EL packets sent by the ingress node are sequentially numbered in their respective packet number space through the *Slice-level Packet Number* field to maintain ordered slice-level packet transmission. This field is also used for efficient packet loss detection at the egress node and caching-based packet retransmission at the ingress node.

3) *Slice-Level VR Video Packet Transmission:* As shown in Fig. 6, based on the designed slice-level packet header, when VR video packets from (remote) servers arrive at the

| | | |
|-----------------|---------------|-------------------------------------|
| Type (i) = 0x05 | Stream ID (i) | Application Protocol Error Code (i) |
|-----------------|---------------|-------------------------------------|

Fig. 7. The STOP_SENDING frame format.

ingress node, header conversion takes place as described above. Intermediate switches along the routing path of the VR video slice transmit packets according to the first-in-first-out (FIFO) principle and directly discard any EL packet that violates its transmission deadline. Once VR video packets reach the egress node, header reversion is performed. Due to possible deadline-violated EL packet dropping, outgoing VR video packets from the egress node may be out-of-order (OOF). To address this, each reverted VR video packet is renumbered by modifying the *Packet Number* field in the QUIC packet header. Specifically, the egress node records for each E2E connection the packet number of the most recently sent packet, i.e., the packet sent with the largest packet number. Then, for each reverted video packet to be transmitted, the modified packet number is the largest packet number recorded plus one.

4) *Video Packet Filtering:* The packet filtering functionality is implemented by leveraging flexible stream-level transmission control provided by the QUIC protocol, where the STOP_SENDING frames are sent by video clients to cease the corresponding stream data transmission of outdated EL tiles, based on the established tile-to-stream mappings in Subsection IV-A. The STOP_SENDING frame format is given in Fig. 7. The *Stream ID* field carries the ID of a stream whose STREAM frames need to be filtered and discarded, and the *Application Protocol Error Code* field contains an application-specified reason for stopping sending/transmitting the STREAM frames of the indicated stream, which is FoV prediction error in the considered 360° VR video streaming case. Since head movements are tracked by HMDs, stream/video tile data transmission cessation is triggered by an HMD to begin the packet filtering operations. The detailed workflow of video packet filtering is shown in Fig. 8 and described as follows:

- When sudden head movements occur and cause FoV prediction errors, the set of outdated EL tiles corresponding to the previously predicted FoV is determined by the HMD. Then, the streams to be ceased are determined based on the established tile-to-stream mappings, and the corresponding STOP_SENDING frames are generated by the HMD. The generated STOP_SENDING frames may be multiplexed with urgent EL tile requests to form uplink urgent request packets. For uplink packet transmission between a pair of egress and ingress nodes, a strict priority-based scheduling policy is applied at the egress node where packets with a higher priority (i.e., the Priority field set to 1) are preemptively scheduled for transmission, while packets with the same priority are scheduled following the FIFO principle [33].
- When the egress and ingress nodes receive the uplink urgent request packets, the Stream IDs in STOP_SENDING frames are extracted for performing packet filtering on received downlink video packets.

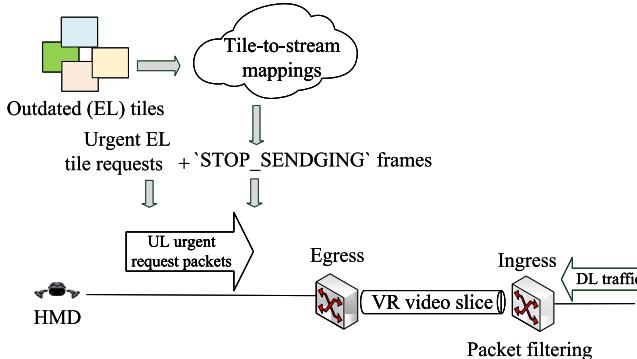


Fig. 8. The workflow of video packet filtering.

Specifically, STREAM frames with the same IDs as those extracted from the STOP_SENDING frames are ruled out from packets upon reception while other (useful) STREAM frames stay.

- When the server receives the urgent request packets, the outdated streams are terminated, and corresponding sending buffers are cleared out. In addition, STREAM frames containing video data of the requested urgent EL tiles are multiplexed and assembled into downlink urgent EL packets, as described in Subsection IV-A.

Note that with the established tile-to-stream mapping relations, operations to specific BL/EL tiles are realized through operating the corresponding streams. Thus, no information from the application layer is required. In addition, en-route video data of outdated EL tiles due to FoV prediction errors is filtered out, and the sending of outdated streams is terminated at the server, in response to viewing behavior dynamics for improving the transmission efficiency.

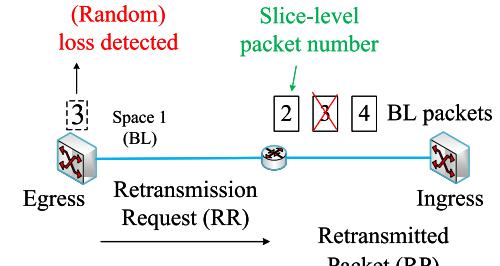
D. Packet Loss Recovery

Differentiated packet loss detection and retransmission mechanisms are required for BL and EL packets, due to their disparate transmission reliability requirements. Owing to the Packet Number Space field in the designed slice-level packet header (see Fig. 5) and the separate packet number space used for BL/EL packet transmission, we design different packet loss recovery schemes for BL and EL packets respectively in the following.

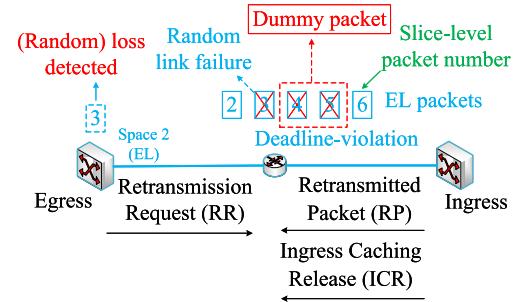
1) *BL Packet Retransmission:* Since BL and EL packets are numbered and sent in order in their respective packet number space, packet loss due to random link failure (referred to as random packet loss) can be detected by the egress node based on the slice-level packet numbers of two consecutively received BL/EL packets where OFO packet arrival indicates a random packet loss.¹

As shown in Fig. 9(a), when a random BL packet loss is detected, the egress node sends a *Retransmission Request (RR)*

¹When a VR video slice between a pair of ingress and egress nodes has a multi-path virtual network topology, a packet scheduling algorithm needs to be implemented at the ingress node where BL and/or EL packets are managed to be sent over different paths according to their discrepancies in terms of link capacity and packet delay, to ensure in-order packet reception at the egress node [34], [35].



(a) BL packet loss detection and retransmission.



(b) EL packet loss detection and retransmission.

Fig. 9. Caching-based BL/EL packet loss recovery.

packet to the ingress node and starts a timer for detecting any RR packet loss. Each RR packet contains three important fields: 1) Flag, a packet type indicator, 2) Packet Number Space, for either BL or EL packets, and 3) *Requested Packet Number*, the slice-level packet number of a lost packet that needs retransmission. This field is in the Optional fields of the designed slice-level packet header (see Fig. 5). When the ingress node receives an RR packet, it retransmits the lost packet using the previously cached packet copy in the caching buffer, as introduced in Subsection III-C.

2) *EL Packet Retransmission:* Since deadline-violated EL packets are directly discarded without retransmissions, the egress node should be equipped with the capability of differentiating between EL packet losses due to random link failure and due to transmission deadline violation. To achieve this, we propose that each intermediate switch generates a dummy packet in place of consecutively discarded EL packets due to deadline violation, as shown in Fig. 9(b). In each dummy packet, the slice-level packet number of the first dropped EL packet is recorded in the Slice-level Packet Number field of the slice-level packet header. The number of consecutively dropped EL packets is put in the Optional fields. Besides, the egress node detects a random EL packet loss via OFO packet reception as in the case of BL packets.

When a random EL packet loss is detected by the egress node, similarly, it sends an RR packet to the ingress node, while EL packet losses due to transmission deadline violation, indicated by dummy packets, are neglected. Then, the ingress node retransmits the lost EL packet if its copy is still stored in the caching buffer upon receiving the RR packet. Note that a lost EL packet is worth retransmission only if its

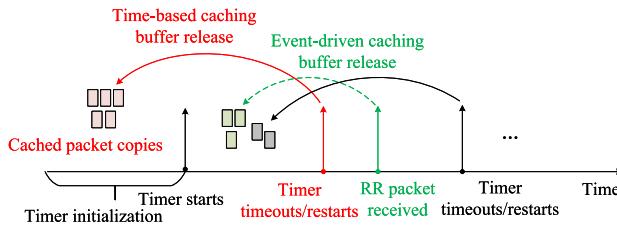


Fig. 10. The workflow of event-driven and time-based caching buffer releases.

transmission deadline is not exceeded. If the requested EL packet copy is already discarded from the caching buffer due to deadline violation, the ingress node responds with an *Ingress Caching Release (ICR)* packet to avoid meaningless retransmission requests from the egress node. In an ICR packet, the smallest slice-level packet number among the cached EL packet copies in the caching buffer is recorded in the optional field of *Minimum (Slice-level) EL Packet Number* in the slice-level packet header. When the egress node receives an ICR packet, all the randomly lost EL packets with slice-level packet numbers smaller than the value indicated in the Minimum (Slice-level) EL Packet Number field no longer need retransmissions.

3) *Caching Buffer Release*: The caching buffer of the ingress node needs to be released regularly to avoid overflow. Consequently, we use event-driven and time-based methods for caching buffer release. For the event-driven method, a received RR packet is used as a trigger to release the cached BL/EL packet copies. Specifically, take BL packets as an example, when the ingress node receives an RR packet requesting retransmission of the BL packet with slice-level packet number i , all the cached BL packet copies with slice-level packet numbers smaller than i are released. For the time-based method, in the case of no random BL/EL packet loss happening in the core network for a certain period of time, a timeout timer is set at the ingress node to periodically release the cached packet copies. In addition, due to the low transmission reliability requirement of EL packets, the ingress node can proactively discard deadline-violated EL packet copies in the caching buffer without notifying the egress node.

The proposed caching-based packet retransmission scheme is efficient in terms of reduced signaling overhead (i.e., ACK frames) for packet loss detection, adaptiveness to disparate transmission reliability requirements, and better usage of link transmission resources due to deadline-violated EL packet dropping with no retransmission.

V. TRANSPORT PARAMETER CALCULATION

The workflow of caching buffer release at the ingress node is shown in Fig. 10, including the event-driven method based on a received RR packet and the time-based method by setting up a periodic timer. Specifically, during the timer initialization phase, the ingress node transmits BL/EL packets received from remote servers on the Internet for the first time and stores their copies in the caching buffer. Then, the ingress node starts the periodic timer while continuing to receive packets and put their copies in the caching buffer. Every time the timer timeouts, the

ingress node releases the packet copies that are stored before the timer starts from the caching buffer. The periodic timer restarts when it timeouts.

There are two key transport parameters to be determined in the proposed caching-based packet retransmission scheme. First, the duration of the timer initialization phase and the cycle of the periodic timer need to be carefully decided, such that the cached packet copies in the caching buffer are released before they are received by the egress node, and the caching resources are efficiently utilized. If a '*release-before-arrival*' event happens to a packet that is randomly lost during the transmission, the lost packet has to be retransmitted from a remote server instead of from the ingress node, leading to a large packet retransmission delay. Second, considering the two caching buffer release methods, a minimum caching buffer size can be derived such that, on average, copies of arriving packets at the ingress node can be put into the caching buffer without causing buffer overflow.² Since each RR packet must be received during a cycle of the periodic timer, the minimum caching buffer size required for the event-driven (caching buffer) release method is smaller than that for the time-based release method. Therefore, in this section, we focus on the periodic caching buffer release (i.e., time-based) and derive the minimum caching buffer sizes (in packet) required for BL and EL packets.

Specifically, we consider the ‘best case’ where there are no deadline-violated and outdated EL packets being discarded during the slice-level VR video packet transmission. In fact, the amount of outdated EL video data to be filtered out and the number of additionally requested urgent EL tiles depend on the FoV prediction error, which affects the aggregated EL traffic arrival rate at the ingress node. Thus, without loss of generality, we consider the aggregated BL and EL traffic of the same VR video slice as two independent Poisson processes [36]. For the time-based caching buffer release method, the minimum required caching buffer size, the duration of the timer initialization phase, and the cycle of the periodic timer should be determined based on the estimated average E2E packet delay for traversing a VR video slice, as discussed in the following.

A. E2E Packet Transmission Analysis

As illustrated in Subsection III-A, we consider that a VR video slice between a pair of ingress (s_0) and egress nodes (s_W) in the core network has a linear topology consisting of $W+1$ nodes interconnected by W transmission links. The link capacity (in packet/s) of node s_i is denoted by μ_i . We denote the aggregated BL and EL packet arrival rates at the ingress node by λ^b and λ^e , respectively.

Packet transmission at the first node (i.e., the ingress node) is modeled as an $M/D/1$ queuing system with the total packet arrival rate $\lambda_0 = \lambda^b + \lambda^e$ and transmission rate μ_0 . The average packet delay at the first node is given by [37]

$$T_0 = \left(1 + \frac{1}{2} \cdot \frac{\rho_0}{1 - \rho_0} \right) \cdot \frac{1}{\mu_0} \quad (1)$$

²Extra caching resources need to be assigned against the burstiness of traffic arrivals, which is out of the scope of this work.

where $\rho_0 = \frac{\lambda_0}{\mu_0}$ is the traffic intensity at the first node, indicating the fraction of time the system is busy.

Next, we focus on the second node. Packet transmission at the second node is closely correlated to its preceding node regarding link transmission rate and packet departure process. Specifically, if the transmission rate of the second node is not less than that of the first node, i.e., $\mu_0 \leq \mu_1$, an arriving packet at the second node is immediately transmitted before the arrival of the next packet. Thus, packet delay at the second node, T_1 , consists of only packet transmission delay, i.e., $T_1 = \frac{1}{\mu_1}$, and packet departure process at the second node remains the same as that at the first node. On the other hand, if $\mu_0 > \mu_1$, newly arriving packets may wait in the second node's transmission queue when a packet is being transmitted. In this case, both packet queuing and transmission delays exist and should be considered.

Let random variable Z_0 be the inter-departure time of successive packets departing from the first node. If a departing packet sees a nonempty queue, then $Z_0 = \frac{1}{\mu_0}$. Otherwise, $Z_0 = \xi + \frac{1}{\mu_0}$, where ξ represents the time interval from the departure of a packet to the next packet arrival. Due to the memoryless property of Poisson arrivals, ξ has the same exponential distribution with parameter λ_0 as the packet inter-arrival time. It is seen that packet departures from the first node follow a mixed process alternating between Poisson and deterministic processes. Due to the level crossing property and the fact that the Poisson Arrivals See Time Averages (PASTA) property holds for an M/D/1 queuing system [37], the steady-state probability of queue length seen by a departing packet is the same as that seen by an arriving packet or at an arbitrary time [38]. Hence, the mean and variance of packet inter-departure time at the first node, Z_0 , are derived as

$$\begin{aligned} E[Z_0] &= \rho_0 \cdot \frac{1}{\mu_0} + (1 - \rho_0) \cdot E\left[\xi + \frac{1}{\mu_0}\right] = \frac{1}{\lambda_0} \\ D[Z_0] &= E\left[(Z_0 - E[Z_0])^2\right] = \frac{1}{(\lambda_0)^2} - \frac{1}{(\mu_0)^2}. \end{aligned} \quad (2)$$

We can observe from Eq. (2) that, when the link transmission rate μ_0 is large, packet departures from the first node approach to a Poisson process. When λ_0 increases to approach μ_0 , i.e., when the queuing system is heavily loaded, packet departures from the first node approach to a deterministic process, which is consistent with the preceding analysis. To achieve the independence between two consecutive transmission nodes for analysis tractability, under the assumption of a large transmission rate, we approximate the packet departures from a node (or the packet arrivals at its subsequent node) as a Poisson process. Therefore, when $\mu_0 > \mu_1$, the packet delay at the second node, T_1 , can be calculated according to Eq. (1) with traffic intensity $\rho_1 = \frac{\lambda_0}{\mu_1}$ and transmission rate μ_1 .

Finally, the E2E (BL/EL) packet delay for traversing a VR video slice in the core network is given by

$$T^E = \sum_{i=0}^{W-1} T_i \quad (3)$$

where T_i is the packet delay for passing through the i -th node along the route of the VR video slice.

For the time-based caching buffer release, the cycle of the periodic timer should be set based on the estimated E2E packet delay (i.e., Eq. (3)). Specifically, the timer cycle should be at least equal to or greater than the estimated E2E packet delay, such that the cached packet copies in the caching buffer are released after the egress node has successfully received them. In that case, any random BL/EL packet loss happening during the slice-level packet transmission is retransmitted from the ingress node instead of from a remote server on the Internet.

B. Minimum Required Caching Buffer Size

Let k_I and k_c be the initialization phase duration and the cycle of the periodic timer, respectively. As shown in Fig. 10, k_I is the time interval that the ingress node transmits packets received from remote servers for the first time. If $k_I < k_c$, in order to ensure that the set of BL packet copies for each periodic caching buffer release is released after they are successfully received by the egress node, the minimum caching buffer size required for BL packets is $\lceil \lambda^b (2k_c) \rceil$. However, the caching resource utilization during the initialization phase is low. If $k_I > k_c$, in order to ensure that the packet copies corresponding to the set of BL packets transmitted by the ingress node during the initialization phase can be stored at the caching buffer and be released after they reach the egress node, the minimum required caching buffer size for BL packets is $\lceil \lambda^b (k_I + k_c) \rceil$. However, in this case, caching resources are not efficiently utilized after the cached packet copies corresponding to the set of BL packets transmitted during the initialization phase are released. Therefore, the optimal initialization phase duration is $k_I^{opt} = k_c$, and the minimum caching buffer size required for BL packets, C^b , is given by

$$C^b = \lceil \lambda^b (2k_c) \rceil, \quad k_c \geq T^E. \quad (4)$$

Similarly, for EL packets, due to the low transmission reliability requirement, deadline-violated EL packets are directly released from the caching buffer. Let d^e be the average transmission deadline of arriving EL packets at the ingress node. Thus, the minimum required caching buffer size for EL packets, C^e , is given by

$$C^e = \lceil \min(\lambda^e (2k_c), \lambda^e d^e) \rceil. \quad (5)$$

The minimum total caching buffer size required for both BL and EL packets is given by

$$C^{min} = C^b + C^e. \quad (6)$$

To verify the performance of the proposed caching-based packet retransmission scheme with the caching buffer size set according to Eq. (6), we consider a VR video slice with a linear topology which consists of 8 transmission nodes including the ingress and egress nodes (i.e., $W = 7$). The link transmission rates are set as [400, 400, 300, 300, 200, 200, 200] in packet/s, which includes both of the cases when there is or is no queuing delay at a node. The BL and EL traffic arrival rates are $\lambda^b = 100$ and $\lambda^e = 50$ in packet/s, respectively. We set the cycle of the periodic timer by the estimated E2E packet delay. The results show that the caching miss ratio

TABLE I
MAIN SIMULATION PARAMETERS

| Parameters | Value |
|--|--|
| The number of video segments | 15 |
| Video segment duration | 2 s |
| The number of encoded packets per video tile | 10 |
| QUIC packet size | 1200 bytes |
| Video tiling layout | 4×8 |
| FoV | $100^\circ \times 100^\circ$ |
| FoV prediction error (in terms of pitch and yaw) | $\mathcal{N}(x^\circ, 10^\circ), x = [0 : 15]$ |
| Link random loss rate | [0.5 : 0.5 : 8]% |
| Congestion interval duration | [0 : 0.5 : 7] s |

caused by ‘release-before-arrival’ packets is 0.83%, which means that most of the cached packet copies are released after they are received by the egress node. A near-optimal packet retransmission delay is achieved, compared to the optimal case when the caching miss ratio is 0. Therefore, the proposed caching-based packet retransmission scheme with the periodic timer cycle and the caching buffer size set based on Eq. (3) and Eq. (6) performs well. Any randomly lost BL/EL packet is highly-likely retransmitted by the ingress node using the corresponding packet copy in the caching buffer.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the effectiveness of our proposed transmission protocol based on real data traces³ [39]. The main simulation parameters are given in Table I.

A. Data Preprocessing

The selected data traces contain viewing orientations of 50 subjects watching ten 360° VR videos from YouTube with Oculus Rift DK2 being the HMD. The ERP is adopted for spherical VR video projection and storage. We choose the ‘Shark Shipwreck’ video and extract its first 30s for experiments. The viewing trajectories are given in radians in terms of yaw (from $-\pi$ to π) and pitch (from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$). In the ERP-formatted 2D video plane, we first conduct sphere-to-plane coordinate transformation based on Eq. (7) and Eq. (8), which maps the yaw (θ_i) and pitch (φ_i) of a spherical video to the horizontal ($w_i \in [0^\circ, 360^\circ]$) and vertical ($h_i \in [0^\circ, 180^\circ]$) coordinates in the 2D video plane.

$$\theta_i = w_i \cdot \frac{2\pi}{W} - \pi, W = 360^\circ \quad (7)$$

$$\varphi_i = h_i \cdot \frac{\pi}{H} - \frac{\pi}{2}, H = 180^\circ. \quad (8)$$

In addition, we consider a 4×8 tiling layout, as shown in Fig. 11. The panoramic video scene is partitioned into 32 video tiles, each of which covers a $45^\circ \times 45^\circ$ (square) view span and is indexed in raster-scan order. We consider an FoV of $100^\circ \times 100^\circ$ and map a specific FoV to the video tile IDs it covers. Finally, according to [40], with a short prediction step (e.g., 3s), FoV prediction errors for either yaw

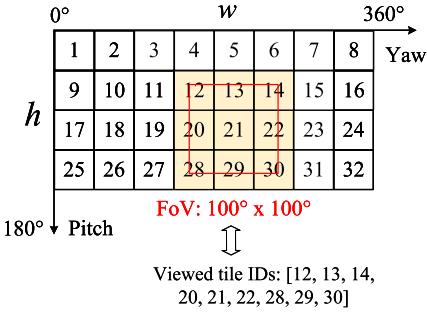


Fig. 11. The considered video tiling layout in the (projected) 2D video plane.

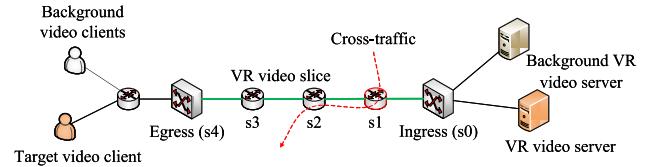


Fig. 12. The considered network scenario in the simulation.

or pitch can be reasonably assumed to follow a Gaussian distribution. Therefore, we add Gaussian noises to the selected data traces to generate the corresponding noised traces with FoV prediction errors for simulation purposes.

B. Simulation Settings

The considered network scenario in the simulation is shown in Fig. 12. Three video clients download video segments from remote VR video servers on the Internet where QUIC are implemented with aioquic.⁴ The aggregated video traffic traverses a VR video slice between a pair of ingress and egress nodes in the core network. Network congestion may happen when different network slices or service traffic pass through the same path and share the transmission resources. In the evaluation, we throttle the link transmission capacity (in packet/s) of intermediate switch s_1 available to the considered VR video slice to represent the case of network congestion due to cross-traffic from other services.

The duration of each video segment is set to 2 seconds [4], and each video segment is encoded into one BL and one EL. Each video client requests video content on a segment basis, and is considered to progressively prefetch 1 video segment ahead only. Specifically, for each video segment downloading, a video client requests full-view BL and only the EL tiles covered by the FoV. In view of the decoding dependency between BL and EL tiles, we consider that BL packets are sent before EL packets by the VR video servers. For simplicity, we assume each video client suddenly rotates head at most once in the middle of watching a video segment. If a video client suddenly rotates head, additional urgent EL tiles corresponding to the updated predicted FoV and the current FoV (after the head rotations) are requested for FoV prediction correction and current viewing experience compensation. The outdated EL tiles corresponding to the previously predicted FoV are filtered out due to viewpoint prediction errors.

³<https://github.com/360VidStr/A-large-dataset-of-360-video-user-behaviour>

⁴<https://github.com/aiortc/aioquic>

C. Performance Metrics

We compare our proposed transmission protocol with QUIC under different network conditions. For brevity, we denote the proposed transmission protocol without deadline-violated EL packet dropping, packet filtering functionality, and caching-based packet retransmission scheme by Proposed wo. dropping, Proposed wo. filtering, and Proposed wo. caching, respectively. Four performance metrics are considered in the evaluation:

- 1) Segment downloading time: The time to receive all the BL tiles of a video segment, indicating the time when a video segment is ready to play smoothly with basic quality;
- 2) Segment quality: The ratio of the number of expected EL tiles timely delivered to the total number of expected EL tiles requested for a video segment, given that the corresponding BL tiles have already been received. Thus, the maximum video segment quality is 2;
- 3) Packet retransmission delay: The waiting time for receiving a retransmitted packet;
- 4) Goodput ratio: The number of received packets that contribute to the video segment quality, including all BL packets and timely-delivered expected EL packets, over the total number of packets received by a video client.

D. Simulation Results

Fig. 13(a) and Fig. 13(b) show the average segment downloading time and quality with link capacity, respectively. As the link capacity of each transmission node increases, the average segment downloading time reduces, and the average segment quality improves, where our proposed transmission protocol achieves better performance compared to the benchmarks. Originally, QUIC is a reliable transport protocol that ensures reliable packet delivery. However, deadline-violated and outdated EL packets do not improve video quality,⁵ which wastes link transmission resources and aggravates network congestion instead. In our proposed transmission protocol, useless EL packets for streaming a video segment are directly discarded, which does not affect the BL/EL packet transmissions of subsequent segments, while link transmission resources are saved to transmit those expected packets that may still be able to be timely delivered to the target video client in a relatively congestion-mild transmission environment. Hence, our proposed transmission protocol achieves better performance than the QUIC protocol.

For Proposed wo. filtering, deadline-violated EL packets are discarded without affecting the BL tile downloading of the next video segment, thus achieving a smaller average segment downloading time than the Proposed wo. dropping. In terms of average segment quality, as link capacity is small, there are many deadline-violated EL packets which account for the majority of useless EL packets. The Proposed wo. filtering scheme thus achieves better average segment quality than the Proposed wo. dropping. As link capacity increases, the number of deadline-violated EL packets is small. In this

⁵We use ‘useless EL packets’ hereafter to refer to deadline-violated and outdated EL packets for brevity.

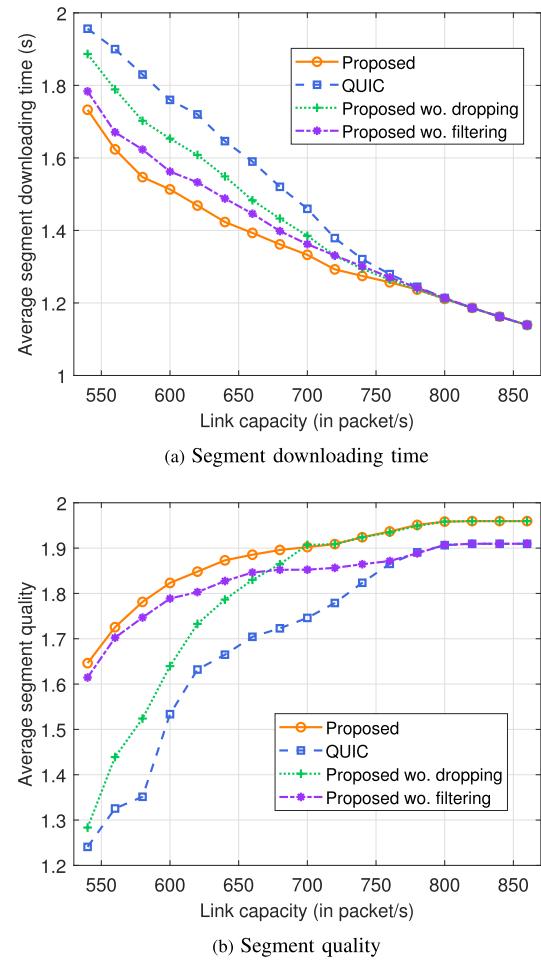


Fig. 13. Average segment downloading time and quality vs. link capacity.

case, filtering out outdated EL packets brings larger segment quality improvement. For Proposed wo. filtering, part of the saved transmission resources is still used to transmit outdated EL packets which account for the majority of useless EL packets when link capacity is large. Therefore, the Proposed wo. dropping scheme achieves higher average segment quality than Proposed wo. filtering.

Fig. 14(a) - Fig. 14(c) show the performance of average packet retransmission delay, average segment downloading time, and average segment quality when random packet loss occurs at each hop along the route of the VR video slice. The mean of FoV prediction error is set as 5°. It can be seen that as random loss rate increases, the average packet retransmission delay and segment downloading time increase, and the average segment quality decreases, while our proposed transmission protocol outperforms the benchmarks. As random loss rate increases, more BL/EL packets are lost and re-injected into the VR video slice for retransmissions. Retransmitted BL packets lead to increased segment downloading time, while retransmitted EL packets degrade segment quality due to deadline violations. With the proposed caching-based packet retransmission scheme, randomly lost BL/EL packets are retransmitted by the ingress node using the cached packet copies in the caching buffer, instead of by the remote servers. Thus, a smaller packet retransmission delay is

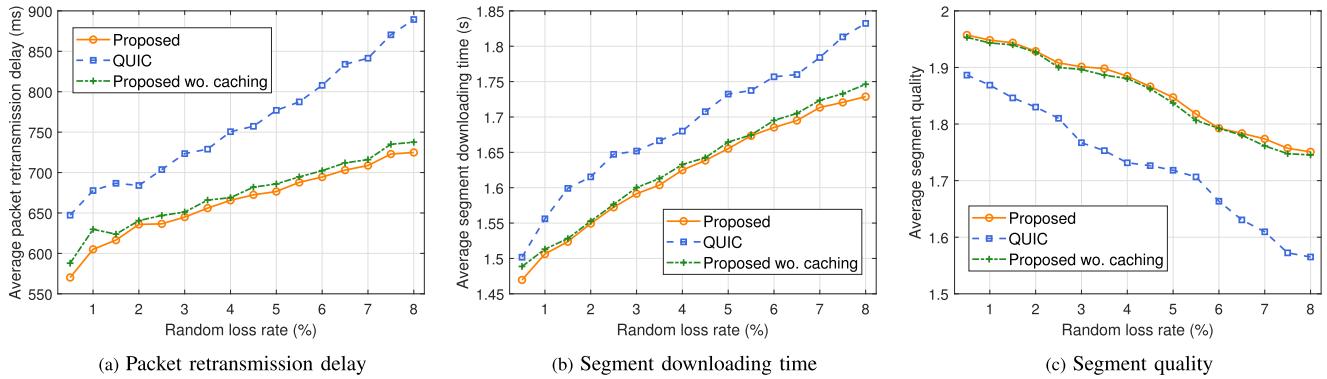


Fig. 14. Average packet retransmission delay, segment downloading time, and segment quality vs. random packet loss rate.

achieved. Besides, deadline-violated and outdated EL packets are discarded without being further transmitted, including the retransmitted ones. On the contrary, in QUIC, all lost BL/EL packets including the deadline-violated ones are retransmitted from the remote servers. Therefore, our proposed transmission protocol outperforms the QUIC protocol. In addition, when the proposed caching-based packet retransmission scheme is deactivated (i.e., Proposed wo. caching), randomly lost BL/EL packets are retransmitted in a less congested transmission environment compared to QUIC since useless EL packets are directly dropped, which, resultingly, achieves better performance.

Fig. 15(a) and Fig. 15(b) show the average segment downloading time and quality with the mean of FoV prediction error. When the mean of FoV prediction error increases, the predicted FoV deviates more from the real FoV, and the corresponding sets of EL tiles covered by the predicted and the real FoVs differ more significantly from each other. Correspondingly, there will be more urgent expected EL packets to be delivered and more outdated EL packets to be filtered out. It can be seen from Fig. 15 that the average segment downloading time increases and the average segment quality reduces as FoV prediction deviation becomes larger, while our proposed transmission protocol outperforms the benchmarks. In the proposed transmission protocol, useless EL packets for each video segment downloading are directly discarded without affecting the subsequent segment's BL/EL packet transmissions, and more urgent expected EL packets can be timely delivered to the target video client. For QUIC, link transmission resources are wasted for reliably transmitting those useless EL packets. Hence, our proposed transmission protocol achieves better performance.

In addition, for Proposed wo. filtering, deadline-violated EL packets are discarded, which thus has minor impacts on the subsequent segment's BL tile downloading. On the other hand, as FoV prediction deviation increases, the number of outdated EL packets is larger. In this case, packet filtering operations obtain bigger segment quality improvement. For the Proposed wo. filtering, part of the saved transmission resources due to deadline-violated EL packet dropping is still consumed to transmit outdated rather than expected EL packets. Therefore, compared to the Proposed wo. dropping,

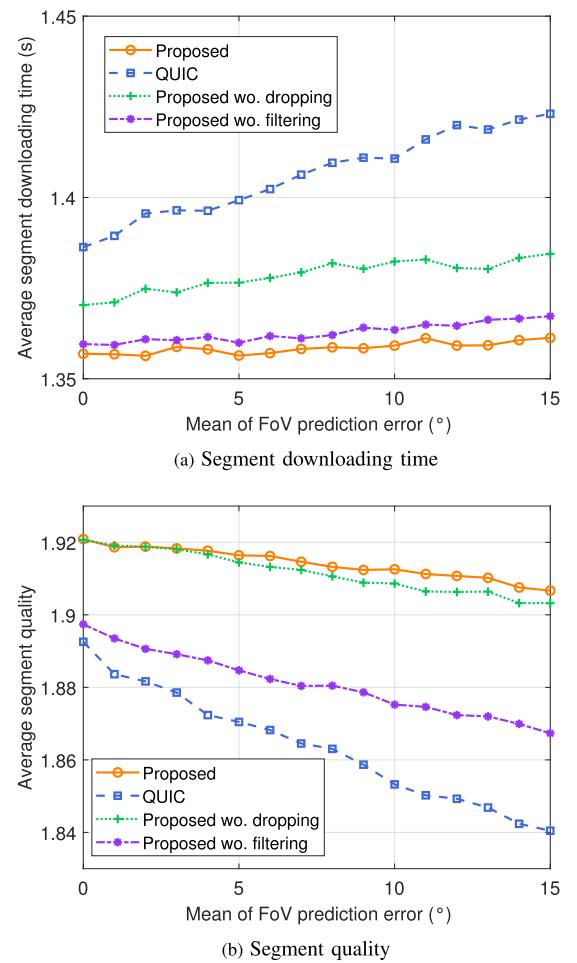


Fig. 15. Average segment downloading time and quality vs. FoV prediction error.

the Proposed wo. filtering scheme achieves a lower average segment downloading time and quality.

Fig. 16(a) - Fig. 16(b) show the average segment download time, quality, and average goodput ratio with congestion duration, respectively. Specifically, starting from the 10th second in the evaluation, we throttle the link transmission rate of s_1 to 540 packet/s. We can see from Fig. 16 that the average segment downloading time increases and the average segment

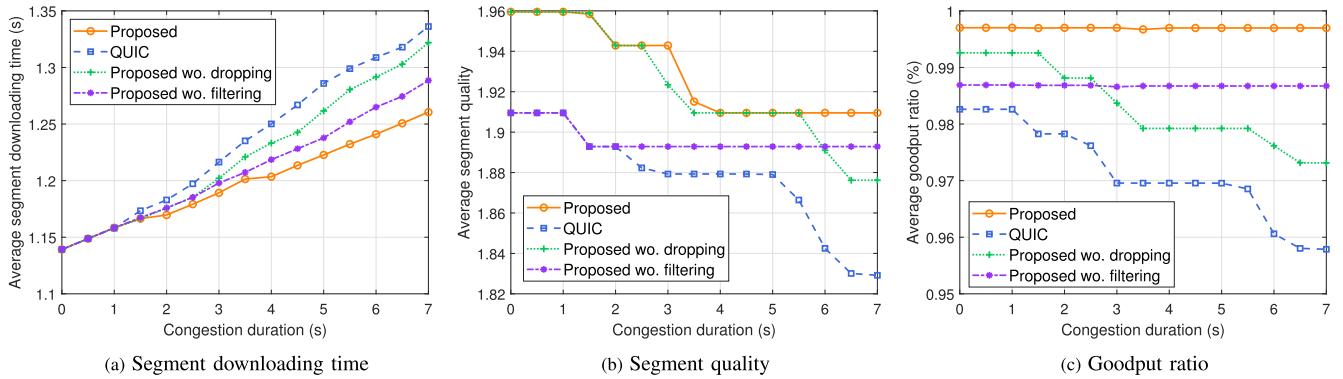


Fig. 16. Average segment downloading time, segment quality, and goodput ratio vs. congestion duration.

quality decreases with congestion duration, and our proposed transmission protocol outperforms the benchmarks, showing a lower sensitivity to network congestion. Besides, our proposed transmission protocol maintains a very high average goodput ratio, indicating high transmission efficiency.

When the transmission network is congested, it takes longer time to receive all the BL tiles of a video segment, and more expected EL packets exceed their transmission deadlines, thus leading to a longer segment downloading time and degraded segment quality. In the proposed transmission protocol, useless EL packets are directly dropped without affecting the subsequent segment's BL/EL packet transmissions. For QUIC, useless EL packets are still reliably transmitted, which aggravates network congestion and hinders the timely delivery of subsequent BL and expected EL packets, thus leading to a larger average segment downloading time and lower average segment quality. In addition, for the Proposed wo. filtering, when congestion duration is short, the number of deadline-violated EL packets is small, and part of link capacity is taken up to timely deliver outdated EL packets rather than urgent expected EL packets. Thus, the Proposed wo. filtering achieves lower average segment quality than the Proposed wo. dropping. When congestion duration increases, the number of deadline-violated EL packets is large, some of which are likely outdated EL packets. In that case, the Proposed wo. filtering achieves higher average segment quality than the Proposed wo. dropping. In addition, goodput ratio loss may come from deadline-violated EL packets when deadline-violated EL packet dropping is not enabled or from outdated EL packets when packet filtering functionality is not activated. With our proposed transmission protocol, almost all the EL packets received by the target video client are expected and timely-delivered, thus maintaining a very high average goodput ratio.

VII. CONCLUSION

In this paper, we have presented a customized slice-level transmission protocol based on QUIC for tile-based 360° VR video streaming. Various properties of video tiles are supported by tailoring the QUIC protocol, where explicit mapping relations between requested video tiles and QUIC streams are established. Two customized protocol functionalities including packet filtering and caching-based packet retransmission are

proposed which remove outdated EL video data due to FoV prediction errors in prompt response to viewing behavior dynamics and efficiently perform retransmissions of lost BL and EL packets with disparate transmission reliability requirements. Key transport parameters are determined via analytical modeling. Simulation results demonstrate the effectiveness of the proposed transmission protocol. As the E2E packet transmission analysis in this work considers the ‘best case’ neglecting the effect of deadline-violated EL packet dropping, we will consider a more realistic situation and develop a more practical E2E transmission modeling in our future work.

REFERENCES

- [1] Y. Wei, Q. J. Ye, K. Qu, W. Zhuang, and X. S. Shen, “Transmission protocol customization for on-demand tile-based 360° VR video streaming,” in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Aug. 2024, pp. 1069–1074.
- [2] S. Z. A. Ansari, V. K. Shukla, K. Saxena, and B. Filomeno, “Implementing virtual reality in entertainment industry,” in *Cyber Intelligence and Information Retrieval*. Berlin, Germany: Springer, 2022, pp. 561–570.
- [3] J. Kim, K. Kim, and W. Kim, “Impact of immersive virtual reality content using 360-degree videos in undergraduate education,” *IEEE Trans. Learn. Technol.*, vol. 15, no. 1, pp. 137–149, Feb. 2022.
- [4] J. V. D. Hooft et al., “A tutorial on immersive video delivery: From omnidirectional video to holography,” *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 1336–1375, 2nd Quart., 2023.
- [5] A. Yaqoob, T. Bi, and G.-M. Muntean, “A survey on adaptive 360 video streaming: Solutions, challenges and opportunities,” *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2801–2838, 4th Quart., 2020.
- [6] Y. Wang, J. Li, Z. Li, S. Shang, and Y. Liu, “Synergistic temporal-spatial user-aware viewport prediction for optimal adaptive 360-degree video streaming,” *IEEE Trans. Broadcast.*, vol. 70, no. 2, pp. 453–467, Jun. 2024.
- [7] N. Kan, J. Zou, C. Li, W. Dai, and H. Xiong, “RAPT360: Reinforcement learning-based rate adaptation for 360-degree video streaming with adaptive prediction and tiling,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 1607–1623, Mar. 2022.
- [8] H. Zhang, Y. Ban, Z. Guo, K. Chen, and X. Zhang, “RAM360: Robust adaptive multi-layer 360 video streaming with Lyapunov optimization,” *IEEE Trans. Multimedia*, vol. 25, pp. 4225–4239, 2023.
- [9] J. Zeng, X. Zhou, and K. Li, “MADRL-based joint edge caching and bitrate selection for multicategory 360-degree video streaming,” *IEEE Internet Things J.*, vol. 11, no. 1, pp. 584–596, Jan. 2024.
- [10] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, “Overview of SHVC: Scalable extensions of the high efficiency video coding standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 20–34, Jan. 2016.
- [11] M. T. Vega et al., “Immersive interconnected virtual and augmented reality: A 5G and IoT perspective,” *J. Netw. Syst. Manag.*, vol. 28, no. 4, pp. 796–826, 2020.

- [12] H. Wang, H. Dong, and A. E. Saddik, "Tile-weighted rate-distortion optimized packet scheduling for 360° virtual-reality video streaming," *IEEE Intell. Syst.*, vol. 39, no. 4, pp. 60–72, Jul. 2024.
- [13] Q. Ye, J. Li, K. Qu, W. Zhuang, X. S. Shen, and X. Li, "End-to-end quality of service in 5G networks: Examining the effectiveness of a network slicing framework," *IEEE Trans. Veh. Technol.*, vol. 13, no. 2, pp. 65–74, Jun. 2018.
- [14] Q. Ye, W. Shi, K. Qu, H. He, W. Zhuang, and X. Shen, "Joint RAN slicing and computation offloading for autonomous vehicular networks: A learning-assisted hierarchical approach," *IEEE Open J. Veh. Technol.*, vol. 2, pp. 272–288, 2021.
- [15] L. Sun et al., "A two-tier system for on-demand streaming of 360 degree video over dynamic networks," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 43–57, Feb. 2019.
- [16] J. Song, F. Yang, W. Zhang, W. Zou, Y. Fan, and P. Di, "A fast FoV-switching DASH system based on tiling mechanism for practical omnidirectional video services," *IEEE Trans. Multimedia*, vol. 22, no. 9, pp. 2366–2381, Sep. 2020.
- [17] H. Shi, Y. Cui, F. Qian, and Y. Hu, "DTP: Deadline-aware transport protocol," in *Proc. 3rd Asia-Pacific Workshop Netw.*, Aug. 2019, pp. 1–7.
- [18] C. Zhou, W. Wu, D. Yang, T. Huang, L. Guo, and B. Yu, "Deadline and priority-aware congestion control for delay-sensitive multimedia streaming," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 4740–4744.
- [19] D. Su, L. Cui, L. Zhang, Y. Suo, and Y. Qiu, "Rate adaptation and block scheduling for delay-sensitive multimedia applications," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 4828–4832.
- [20] H. K. Ravuri, M. T. Vega, J. D. Van Hooft, T. Wauters, and F. De Turck, "Adaptive partially reliable delivery of immersive media over QUIC-HTTP/3," *IEEE Access*, vol. 11, pp. 38094–38111, 2023.
- [21] C. Cui, Y. Lu, S. Li, J. Li, and Z. Ruan, "DASH+: Download multiple video segments with stream multiplexing of QUIC," in *Proc. 10th Int. Conf. Adv. Cloud Big Data (CBD)*, Nov. 2022, pp. 66–72.
- [22] S.-C. Yen, C.-L. Fan, and C.-H. Hsu, "Streaming 360° videos to head-mounted virtual reality using DASH over QUIC transport protocol," in *Proc. 24th ACM Workshop Packet Video*, Jun. 2019, pp. 7–12.
- [23] N. McKeown et al., "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.
- [24] Y. Lu, Z. Ling, S. Zhu, and L. Tang, "SDTCP: Towards datacenter TCP congestion control with SDN for IoT applications," *Sensors*, vol. 17, no. 1, p. 109, Jan. 2017.
- [25] Y. Xu et al., "RAPID: Avoiding TCP incast throughput collapse in public clouds with intelligent packet discarding," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 8, pp. 1911–1923, Aug. 2019.
- [26] M.-H. Wang, L.-W. Chen, P.-W. Chi, and C.-L. Lei, "SDUDP: A reliable UDP-based transmission protocol over SDN," *IEEE Access*, vol. 5, pp. 5904–5916, 2017.
- [27] J. Chen et al., "SDATP: An SDN-based traffic-adaptive and service-oriented transmission protocol," *IEEE Trans. Cognit. Commun. Netw.*, vol. 6, no. 2, pp. 756–770, Jun. 2020.
- [28] S. Yan, Q. Ye, and W. Zhuang, "Learning-based transmission protocol customization for VoD streaming in cybertwin-enabled next-generation core networks," *IEEE Internet Things J.*, vol. 8, no. 22, pp. 16326–16336, Nov. 2021.
- [29] F. Zou, Y. Wang, and Y. Liu, "A multipath routing approach for tile-based virtual reality video streaming based on SDN," in *Proc. IEEE 45th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2021, pp. 560–565.
- [30] J. Iyengar and M. Thomson, "QUIC: A UDP-based multiplexed and secure transport," RFC Editor, p. 151, May 2021. [Online]. Available: <https://www.rfc-editor.org/info/rfc9000>
- [31] A. Langley et al., "The QUIC transport protocol: Design and internet-scale deployment," in *Proc. Conf. ACM Special Interest Group Data Commun.*, 2017, pp. 183–196.
- [32] B. Davie and J. Gross. (Apr. 2016). *A Stateless Transport Tunneling Protocol for Network Virtualization (STT)*. Internet Engineering Task Force. [Online]. Available: <https://datatracker.ietf.org/doc/draft-davie-stt/08/>
- [33] W. Bai, L. Chen, K. Chen, D. Han, C. Tian, and H. Wang, "PIAS: Practical information-agnostic flow scheduling for commodity data centers," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 1954–1967, Aug. 2017.
- [34] W. Wei, J. Han, Y. Xing, K. Xue, J. Liu, and R. Zhuang, "MP-VR: An MPTCP-based adaptive streaming framework for 360-degree virtual reality videos," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2021, pp. 1–6.
- [35] W. Yang, P. Dong, L. Cai, and W. Tang, "Loss-aware throughput estimation scheduler for multi-path TCP in heterogeneous wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 5, pp. 3336–3349, May 2021.
- [36] M. Lecci, M. Drago, A. Zanella, and M. Zorzi, "An open framework for analyzing and modeling XR network traffic," *IEEE Access*, vol. 9, pp. 129782–129795, 2021.
- [37] D. Bertsekas and R. Gallager, *Data Networks*. Nashua, NH, USA: Athena Scientific, 2021.
- [38] Q. Ye, W. Zhuang, X. Li, and J. Rao, "End-to-end delay modeling for embedded VNF chains in 5G core networks," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 692–704, Feb. 2019.
- [39] W.-C. Lo, C.-L. Fan, J. Lee, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu, "360 video viewing dataset in head-mounted virtual reality," in *Proc. 8th ACM Multimedia Syst. Conf.*, Jun. 2017, pp. 211–216.
- [40] L. Xie, Z. Xu, Y. Ban, X. Zhang, and Z. Guo, "360ProbDASH: Improving QoE of 360 video streaming using tile-based HTTP adaptive streaming," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 315–323.



Yannan Wei (Graduate Student Member, IEEE) received the B.S. and M.S. degrees in information and communication engineering from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research interests include transmission protocol design and resource management for future networking and applications.



Qiang (John) Ye (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Waterloo, ON, Canada, in 2016. Since September 2023, he has been an Assistant Professor with the Department of Electrical and Software Engineering, Schulich School of Engineering, University of Calgary, AB, Canada. Before joining UCalgary, he was an Assistant Professor with the Department of Computer Science, Memorial University of Newfoundland, NL, Canada, from September 2021 to August 2023, and the Department of Electrical and Computer Engineering and Technology, Minnesota State University, Mankato, USA, from September 2019 to August 2021. He was with the Department of Electrical and Computer Engineering, University of Waterloo, as a Post-Doctoral Fellow and then a Research Associate, from December 2016 to September 2019. He has published over 70 research papers in top-ranked journals and conference proceedings. He is/was the general, publication, and program co-chair for different reputable international conferences and workshops. He also serves/served as the IEEE Vehicular Technology Society (VTS) Region 7 Chapter Coordinator in 2024 and the Regions 1–7 Chapters Coordinator from 2022 to 2023. He received the Best Paper Award in the IEEE/CIC International Conference on Communications in China (ICCC) in 2024 and IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING Exemplary Editor Award in 2023. He serves/served as an Associate Editor for prestigious international journals, such as IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING.



Kaige Qu (Member, IEEE) received the B.Sc. degree in communication engineering from Shandong University, Jinan, China, in 2013, the joint M.Sc. degree in integrated circuits engineering and electrical engineering from Tsinghua University, Beijing, China, and KU Leuven, Leuven, Belgium, in 2016, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2021. From February 2021 to December 2023, she was a Post-Doctoral Fellow and then a Research Associate

with the Department of Electrical and Computer Engineering, University of Waterloo. Her research interests include connected and autonomous vehicles, network intelligence, network virtualization, and digital twin-assisted network automation.



Weihua Zhuang (Fellow, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from Dalian Maritime University, China, and the Ph.D. degree in electrical engineering from the University of New Brunswick, Canada.

Since 1993, she has been a Faculty Member with the Department of Electrical and Computer Engineering, University of Waterloo, Canada, where she is currently a University Professor and a University Research Chair. Her current research interests include network architecture, algorithms and protocols, and service provisioning in future communication systems. She is a Fellow of the Royal Society of Canada (RSC), Canadian Academy of Engineering (CAE), and the Engineering Institute of Canada (EIC). She is the President and an Elected Member of the Board of Governors (BoG) of the IEEE Vehicular Technology Society. She was a recipient of the Women's Distinguished Career Award from the IEEE Vehicular Technology Society in 2021, the R.A. Fessenden Award from IEEE Canada in 2021, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2021, and the Technical Recognition Award in Ad Hoc and Sensor Networks from the IEEE Communications Society in 2017. She was the Editor-in-Chief of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY from 2007 to 2013, an Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2005 to 2009, the General Co-Chair of the IEEE/CIC International Conference on Communications in China (ICCC) 2021, the Technical Program Committee (TPC) Chair/the Co-Chair of the IEEE Vehicular Technology Conference 2017 Fall and 2016 Fall, the TPC Symposia Chair of the IEEE Globecom 2011, and an IEEE Communications Society Distinguished Lecturer from 2008 to 2011.



Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, the Internet of Things, AI for networks, and vehicular networks. He is a Registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Member, an International Fellow of the Engineering Academy of Japan, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society. He received the "West Lake Friendship Award" from Zhejiang Province in 2023, the President's Excellence in Research from the University of Waterloo in 2022, Canadian Award for Telecommunications Research from Canadian Society of Information Theory (CSIT) in 2021, the R. A. Fessenden Award from IEEE, Canada in 2019, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award from the IEEE Vehicular Technology Society in 2018, the Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society (ComSoc), and the Technical Recognition Award from the Wireless Communications Technical Committee in 2019 and the AHSN Technical Committee in 2013. He also received the Excellent Graduate Supervision Award from the University of Waterloo in 2006 and the Premier's Research Excellence Award (PREA) from the Province of Ontario, Canada, in 2003. He is the Past President of the IEEE Communications Society. He was the Vice President for Technical and Educational Activities, the Vice President for Publications, the Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a Member of the IEEE Fellow Selection Committee of the ComSoc. He served as the Editor-in-Chief of IEEE INTERNET OF THINGS JOURNAL, IEEE Network, and IET Communications.