Research Repository UCD

Title	QoEMultiSDN: Management of Multimedia Services using MPTCP/SR in Softwarized and Virtualized Networks
Authors(s)	Barakabitze, Alcardo Alex, Mkwawa, Is-Haka, Hines, Andrew, et al.
Publication date	2020-11-23
Publication information	Barakabitze, Alcardo Alex, Is-Haka Mkwawa, Andrew Hines, and et al. "QoEMultiSDN: Management of Multimedia Services Using MPTCP/SR in Softwarized and Virtualized Networks." IEEE, November 23, 2020. https://doi.org/10.1109/ACCESS.2020.3039953.
Publisher	IEEE
Item record/more information	http://hdl.handle.net/10197/25899
Publisher's version (DOI)	10.1109/ACCESS.2020.3039953

Downloaded 2026-05-01 23:43:25

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd_oa)

© Some rights reserved. For more information

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.20xx.DOI

QoEMultiSDN: Management of Multimedia Services using MPTCP/SR in Softwarized and Virtualized Networks

ALCARDO ALEX BARAKABITZE[1,2] (Member, IEEE), IS-HAKA MKWAWA[1] (MEMBER, IEEE), ANDREW HINES[2] (SENIOR MEMBER, IEEE), LINGFEN SUN[1] (MEMBER, IEEE), EMMANUEL IFEACHOR[1] (SENIOR MEMBER, IEEE).

[1]School of Engineering, Computing and Mathematics, University of Plymouth, Drake Circus, PL4 8AA, UK (e-mail: alcardoalex. barakabitze, ishaka.mkwawa, L.Sun, E.Ifeachor@plymouth.ac.uk)
[2]School of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland, (e-mail: andrew.hines@ucd.ie)

Corresponding authors: Alcardo Alex Barakabitze and Lingfen Sun (e-mail: alcardoalex.barakabitze, L.Sun@plymouth.ac.uk).

The work presented in this paper is fully funded by the European Union in the context of Horizon2020 Research and Innovation Programme under Marie Sklodowska-Curie Innovative Training Networks (MSCA-ITN-2014-ETN), Grant Agreement No.643072, Network QoE-NET. The work is also emanated from research supported by the Irish Research Council (IRC) Postdoctoral Fellowship Programme 2019 - 2021 under the project: Management of Multimedia Services and Orchestration of Resources in 5G Networks using SDN and NFV (5GMOONS) and in part by a research grant from Science Foundation Ireland (SFI) and cofunded under the European Regional Development Fund under Grant Number 13/RC/2077 and Grant Number 12/RC/2289_P2.

ABSTRACT 5G networks are set to empower new user experiences for multimedia streaming services by offering high data rates and low latency. The rapid growth of multimedia services, such as video streaming services over future softwarized 5G networks, poses great challenges for both service providers and network operators regarding service provisioning with high Quality of Experience (QoE). Network evolution towards Software Defined Networking (SDN) and Network Function Virtualization (NFV) provides new opportunities to develop innovative ways to address the challenges. In this paper, we present a novel QoE-aware management scheme based on SDN/NFV by utilizing Multi-Path-TCP/Segment-Routing (MPTCP/SR) techniques and exploring information from both the network and client. The new scheme, entitled QoEMultiSDN, provides an optimized end-to-end QoE for multimedia service delivery. QoEMultiSDN employs two novel strategies to achieve this: (a) QoE-based multisource routing and QoE optimization, and (b) multipath protection and dynamic link-failure-free mechanisms. We implemented the proposed scheme over SDN through Dynamic Adaptive Streaming over HTTP (DASH) experiments using Mininet, POX and OpenDaylight controllers, and compared its performance with MPTCP and regular TCP. Experimental results indicate that the QoEMultiSDN outperforms others in terms of system throughput, failure recovery time and the QoE of the end-user.

INDEX TERMS SDN, NFV, QoE, MPTCP, Segment Routing, Multimedia Services and Applications, 5G Networks

I. INTRODUCTION

The consumption of multimedia services has increased considerably over the past few years and this is expected to grow further in the coming years [1]. The latest Cisco Visual Networking Index (VNI) [2] predicts that IP video traffic will represent 82% of all IP traffic by 2022, up from 73% in 2017. Mobile video traffic is expected to account for 79% of mobile data traffic by 2022, a 9-fold increase from 2017. The growth in video traffic is mainly driven by widespread applications, especially mobile applications in video streaming and video gaming services, such as Netflix, YouTube and Twitch. There has been a strong growth in video streaming during the Coronavirus disease (COVID-19) pandemic as more people spend time at home because of lockdown. Advances in networks (e.g. 5G mobile networks, home broadband and the core Internet), video compression (e.g. from H.264 to H.265) and smart devices (e.g. smartphones and smart TV) have contributed to this growth. Developments in 5G mobile technologies [3] and the future Internet will underpin further growth of video traffic and video services, such as immersive/4K/8K video, Augmented Reality (AR), Virtual Reality (VR) or Extended Reality (XR) services over the Internet.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

It is widely accepted that the Quality of Experience (QoE) of the customer is an important factor for the success of the multimedia services [4], [5].

However, providing high QoE services to meet customers' needs is a challenging task for service providers and mobile network operators due to uncertain network conditions, varying user terminals (e.g., computational power/resources, storage capabilities and screen size) and differing requirements from different multimedia services [1]. The other factor is the fast growth and popularity of the multimedia services over the Internet combined with the operational cost optimization by mobile and service providers [1]. The increasing users' demand of multimedia services with better quality have triggered mobile operators and service providers to upgrade their systems and invest in network softwarization and virtualization technologies that leverage the Software Defined Networking (SDN) and Network Function Virtualization (NFV) [3]. SDN and NFV have proven to be effective in providing solutions for traffic management and service provisioning.

Notable advantages of the SDN architecture include centralized control and management, enhanced network programmability, increased network reliability and flexibility as well as data flow optimization [1], [6]. SDN offers a virtualized control plane where intelligent management decisions can be enforced among network functions bridging the gap between service provisioning and QoE management [7]. The network control and management plane in SDN becomes directly programmable using standardized Southbound Interfaces (SBI) such as OpenFlow [8], FoRCES [9] and OpFlex [10]. NFV provides the needed flexibility for Internet Service Providers (ISPs) to support changing business requirements and decrease the deployment time of new network services to market. It also enables ISPs to reduce their OPerational EXpenditure (OPEX) and CAPital EXpenditure (CAPEX) through lower-cost flexible network infrastructures. Major service providers such as Google and Microsoft have proposed their own SDN/NFV solutions, such as Google's B4 SDN [11] and Microsoft SWAN [12]. SDN provides an efficient way of decoupling the data forwarding equipments and network control.

Using SDN, it is possible to automate many tasks within the network (e.g., routing), especially in topologies implemented like a FatTree. As an outcome, SDN-based routing [13] is beneficial for QoE optimization such that, subject to resource constraints and network topology changes, the aggregated user-expected QoE value in a network domain can be achieved [14], [15], [16]. To cope with the increasing growth of video streaming services that require optimal throughput, Multi-Path Transmission Control Protocol (MPTCP) [17], [18] and advanced forwarding technologies such as Segment Routing (SR) [19] have been proposed. MPTCP is an ongoing effort of the Internet Engineering Task Force's (IETF) Multipath TCP working group that aims to maximize system throughput and network resources utilization by using multiple paths transmission [20], [21], [22]. MPTCP was proposed as an extension of TCP with the main idea of transmitting network traffic (e.g., video flows) using multiple numbers of available paths in the network. MPTCP is implemented in the transport layer and supports both IPv4 and IPv6. The data packets that originate from different paths are aggregated at the receiving side which also supports MPTCP. MPTCP avoids network hot-spots by distributing traffic flows over multiple concurrent paths such that all the network links are optimally loaded. Moreover, the network performance of applications can be boosted using MPTCP by aggregating bandwidth over multiple paths using subflows that originates from the same TCP connection [17]. However, MPTCP has three drawbacks including: (1) it lacks dynamic control over selecting the number of subflows to achieve a maximum system throughput; (2) it is a transport layer protocol, with no control over the actual transmission paths taken in the network. This can limit the performance of MPTCP when subflows are transmitted through the same path, and (3) a large number of out-of-order TCP packets may cause poor performance of MPTCP when the paths have different delays and bandwidths.

However, the performance of MPTCP in the network can be affected because it lacks an intelligent mechanism that can control and route the subflows in different paths based on some network constraints such as link bandwidth, packet loss and delay [18]. In that respect, the SDN controller can be utilized to provide a control mechanism and route efficiently the MPTCP subflows in the network. It is worth mentioning that, the SDN controller that has the global view of the network can provide a programmable environment where intelligent QoE-based multipath routing mechanisms for the MPTCP subflows can be implemented [23]. SDN applies flow-based forwarding rules in switches to provide a finer control of the network traffic when transmitting data from source to destination. SDN switches need Ternary Content Addressable Memories (TCAMs) to store routing tables. However, TCAMs are expensive, limited in size and need high power consumption [24]. Stephens et al. [25] demonstrate that, TCAMs support no more than about 25 thousands of 12-tuples forwarding rules. TCAMs introduce a fundamental trade-off between forwarding table size and cost and power incurred in SDN networks. The maximum size of routing tables in switches represents an important challenge for the deployment of video streaming services in SDN. Most of the existing works [26], [17], [27], [20] that implement MPTCP and Dynamic Adaptive Streaming over HTTP (DASH) for improving the end-user's perceived QoE rely on installing forwarding rules in the data plane. The drawback of SDN switches to store a large number of forwarding rules for video streaming services can be eliminated using SR [28]. Segment routing is a scalable and flexible way of doing source routing in the network where the source node selects a path and encodes it in the packet header as an ordered list of segments. SR is an effort of the IETF's Source Packet Routing in Networking (SPRING) working group that aims to simplify the Traffic Engineering (TE) management of the control plane.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

With SR, SDN switches do not need to maintain per-demand routing information during video streaming.

In this paper, we provide original practical TE management mechanisms by employing SR and MPTCP in softwarized networks. We aim to speed up the transfer of a large number of multimedia services and facilitate efficient multipath routing between end-points using the SR paradigm. We propose a novel QoE-aware MPTCP/SR-SDN/NFV-based scheme (QoEMultiSDN) to achieve an end-to-end QoE optimization by utilizing both network and client information. We apply the Constrained Shortest Path Model (CSPM) and Multi-Flow Commodity Model (MFCM) to select important intermediate nodes for MPTCP subflows transmission through multiple disjointed shortest paths. We choose the DASH video streaming service as a vehicle to demonstrate the concept of the proposed scheme. This paper is an extension of our previous works [21], [29], [30] where we demonstrated that utilizing MPTCP/SR can improve network resources utilization and the end-user's QoE. To the best of our knowledge, this is the first paper that improves the end-user's video quality in SDN/NFV networks while taking into accounts multipath protection and dynamic recovery of link failures using MPTCP and SR. The main contributions of this paper are threefold.

- First, we propose the QoEMultiSDN scheme that can achieve an optimized QoE control and management of video streaming services in softwarized networks. MPTCP and SR are employed to forward the network traffic using multiple disjointed paths and utilize efficiently the available network resources.
- Second, we formulate a QoE-based multipath source routing and quality optimization algorithm that can forward network traffic using SR paradigms over the proposed SDN/NFV system.
- Third, we present a multipath protection and dynamic link-failure-free algorithm that greatly reduces failure recovery time and avoids link congestion in MPTCP/SR-based SDN/NFV networks.

We implemented the proposed scheme and relevant algorithms over the developed softwarized - DASH based platform using Mininet, POX and OpenDaylight controllers. We evaluated QoEMultiSDN and compared its performance with existing approaches such as MPTCP and regular TCP. The experimental results indicate that, the QoEMultiSDN performs better in terms of system throughput, failure recovery time and video quality when compared to MPTCP/TCP approaches.

The rest of this paper is organized as follows: Section II reviews related work in the areas of QoE provisioning in SDN-based environments using MPTCP and/or SR. Section III describes the MPTCP and SR paradigms including the required components and operations. Section IV outlines the proposed QoEMultiSDN architecture. We present the proposed QoE-driven multipath source routing and quality optimization, as well as multipath protection and dynamic link-failure-free algorithms in Section V. Section VI, provides the evaluation environment and the results of performance comparison. Finally, Section VII concludes the paper.

II. RELATED WORK

A. MPTCP SDN/NFV-BASED APPROACHES FOR VIDEO STREAMING

Sonkoly et al. [31] introduce an SDN-based control and measurement framework where MPTCP experiments can be conducted to evaluate the concept of multiple paths on top of GEANT and PlanetLab Europe. Kukreja et al. [23] implement an MPTCP manager by configuring switches to forward subflows over different paths using an Openflow-based SDN controller. To maximize the downloading rates and improve the end-users' QoE, Nam et al. [17] build a dynamic MPTCP path control in an SDN-enabled network. The authors adjust the number of MPTCP paths based on the available link capacity to reduce a large number of out-of-order packets which may cause poor performance and degrade the end-user's QoE. Jingpu et al. [32] employ a centralized controller for intelligent subflow route calculation for each MPTCP session in SDN-based data centers. To enable load balancing, Kalpana et al. [33] propose a SubFlow Optimizer (SFO) that establishes new subflows and assigns them to the best paths in SDN. These approaches can increase the load and overhead at SDN switches and the controller because they mainly rely on installing a larger number of rules in the data plane. Also, some of the proposals do not provide path protections and dynamic link recovery strategies during multimedia services delivery in SDN-based networks. This can degrade the end user's QoE when a link/node fails in the network. Such limitations can be easily eliminated using the SR technology when combined with MPTCP for data transmission in SDN/NFV-based networks. As pointed out by Barakabitze et al. [21], SR reduces the number of forwarding rules in SDN/NFV networks. This is because, forwarding elements (e.g., switches) on the transmission path do not need to have a flow rule entry for a traffic-demand for forwarding data packets to the next hop in the network [29].

B. SEGMENT ROUTING SDN/NFV-BASED APPROACHES

Segment routing can reduce the number of forwarding rules in SDN switches by encoding TE paths into SR paths. Trimponias et al. [41] use SR and node centrality concepts from graph theory to select a few important nodes as middle points to route all traffic in SDN-based Wireless Access Networks (WANs). The authors in [40] propose an OpenFlow SR SDN-based strategy that approximately reduces by 50% the number of control messages between source and destination nodes. Luoet al. [24] propose Fast and Lightweight SDN Updates (FLUS), a SR- based mechanism that can handle more than 92% flow updates and reduce up to 88% forwarding rules overhead to the controller. The capabilities provided by SR to reduce the number of forwarding rules in SDN switches give us a motivation to propose a novel strategy

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

TABLE 1: A comparison of MPTCP-based adaptive video streaming solutions and QoEMultiSDN.

Ref-Strategy	DASH-based Approach	QoE-aspects	Features/Asserts	Traffic Load	SDN/NFV-based
[27], ADMIT [20], [26]	Yes	Yes	Improve the end-users' QoE based on real-time video streaming using MPTCP. However, no SDN and/or NFV is considered in their implementations.	High/Medium	NO
[23], [33], [34], [35], [36], [37], [38], [31], [39], [17] & [32]	Yes	Yes	Use of MPTCP to forward traffic through multiple disjointed paths in SDN and/or NFV. It is important to note that, some of these approaches do not consider the aspect of improving the end-user's video quality in real-time video transmissions. Again, these works lead to increased overhead and the deployment cost because of relying on installing a large number of flow rules in SDN switches.	High/Medium	Yes
[24], [40], [41]	No	No	Reduce the number of flow rules in the forwarding elements by encoding TE paths into SR paths. However, no QoE considerations or the application of MPTCP in their implementations	High/Medium	Yes
QoEMultiSDN	Yes	Yes	Improve the end-user's QoE using real-time video transmissions in SDN/NFV environments. Our novel approach provides multiple disjointed shortest paths for multimedia subflows and performs source routing using SR paradigm. We also consider multipath protection and dynamic link recovery mechanisms in SDN/NFV networks.	Low	Yes

called QoEMultiSDN that achieves an efficient QoE control and management of future video streaming services. The proposed architecture can also provide multipath protections and link-failure-free MPTCP/SR-based mechanisms that can increase survivability, resilience, availability and robustness of softwarized networks. Table 1 provides a comparison of QoEMultiSDN architecture and other approaches in the literature.

III. MPTCP AND SEGMENT ROUTING OVER SDN/NFV-BASED NETWORKS

A. THE BASIC OPERATION OF MPTCP IN SDN/NFV-BASED NETWORKS

An MPTCP implementation over SDN/NFV-based networks is illustrated in Fig. 1. We assume an MPTCP connection consisting of 3 MPTCP subflows: *sf1:1*, *sf1:2*, *sf1:3*. The number of subflows is indicated by the indexes and *sf* is the subflow of an MPTCP connection. The SDN controller is responsible to check the available link capacity of all connected paths in the network. It then chooses the shortest paths selected for routing the subflows of a particular MPTCP connection from the source to their destination. Note that, the black boxes in Fig. 1 are the data frames generated as data units by the application during video subflow transmissions. The grey boxes indicate the TCP packets that are received at the client-side. When one packet is lost during transmission, it is re-transmitted. As shown in Fig 1, *sf2* will be transmitted via path S1→S7→S8→S4, and *sf1* via a path, S1→S2→S3→S4. Based on the list of segments, the SR approach can map these subflow paths into SR paths where there is no need to maintain the path state or install flow rules in every SDN switch. The next subsections present the operations of SR and procedures of mapping MPTCP subflow paths into SR paths.

B. TRAFFIC ENGINEERING WITH SEGMENT ROUTING IN SDN

SR is a source routing approach that provides an advanced packet forwarding mechanism where intermediate nodes (routers/switches) are not required to install flow rules or maintain all steered paths within the network. The SR relies on processing/forwarding functions called segments to steer a packet in the network. A segment is a physical or a logical network element such as a network link, an SDN switch or a router. Every segment has an Identifier (ID) which can be of local or global significance. Local Segment routing IDentifiers (SIDs) are only known to the node associated with the SID [42] while global SIDs are known by all nodes in the network. A chain of these elements is formed by a list of SIDs where packets are routed within an SDN/NFV system [21].

C. PATH PROTECTION WITH SEGMENT ROUTING IN SDN/NFV-BASED NETWORKS

Future softwarized networks should be robust enough to ensure high reliability and availability of services by making sure that any failed parts in the network are detected, restored and recovered within a permissible time and at the lowest achievable cost. Path restoration and protection are the most commonly used failover approaches in SDN and NFV [43]. For path protection strategies [44], [45], [46], backup paths are proactively (pre-installed) configured. That way, backup paths are activated automatically in SDN switches when a link fails. With path restoration approaches [43], [47], backup paths are reactively installed when the link/node failure is detected by the SDN controller in the network. However, the failure recovery time using path restoration is larger because the SDN controller has to (a) detect the failed link/node, (b) recompute a new routing path, and (c) install flow entries in the OpenFlow switches directly connected to the failed link.

The path restoration and protection strategies which involve the SDN controller and OpenFlow switches would lead to an increased failure recovery time and a large number of flow rules in SDN switches. Again, for video streaming services, congestion can occur in the network because of redirecting a large number of video flows on the failed link to a new reroute path [45]. Previous studies have shown that the path restoration can take a recovery time of over 100 *ms*, which is longer than 50 *ms* tolerable time considered by a a network provider [46]. The drawbacks of path restoration and protections approach to store a large number of rules in SDN switches can be greatly eliminated by SR [28] technology. As mentioned earlier, SR reduces the number

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

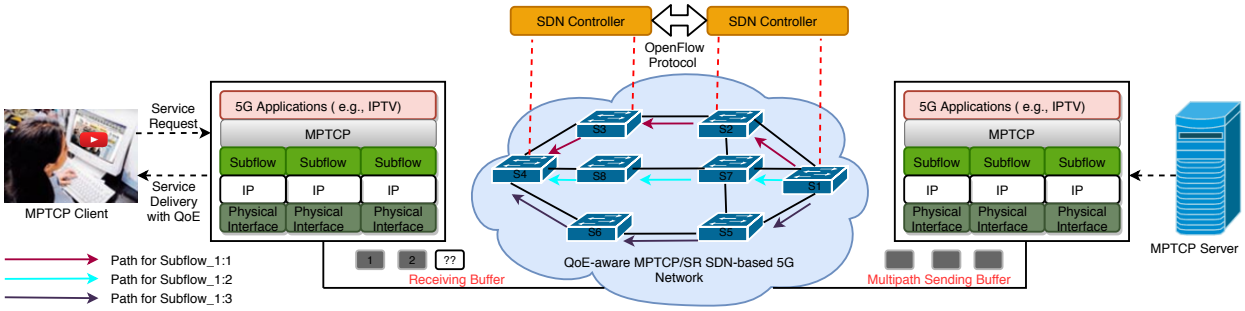


FIGURE 1: A Video Streaming Scenario using MPTCP in SDN-based 5G Networks.

of rules by expressing TE paths into SR paths using a list of segment labels. Path protection and dynamic network recovery from failure is an important aspect of QoE control and management in SDN/NFV networks. This is so because network failures (due to a failed link or node) directly affect transmission quality and ability of service providers to meet their Experience Level Agreements (ELAs)[1]. When network failures occur in the network, segments can be repaired such that new backup paths are computed for each link in the primary path. The scope of this paper is mainly on link failures using SR. With reference to Fig. 1, when link S2→S3 fails, the topology is updated by the SDN controller which has the global knowledge of the network. In such case, link S2→S3 is pruned before new backup paths are computed. A failover table is created where *backup actions* for each source-destination points are also indicated. The controller encodes the required segment list on the ingress-node following the new computed backup paths. The segment list can be used for video transmission from source to destination.

IV. QOEMULTISDN: THE PROPOSED QOE-AWARE MPTCP SDN/NFV-BASED SR ARCHITECTURE

This section presents a novel QoEMultiSDN architecture shown in Fig 2 that achieves an optimized end-to-end QoE level for the end-users. It also provides an efficient QoE control and management of video streaming services in future softwarized networks. The proposed architecture performs source routing using SR paradigm and transmits MPTCP subflows using multiple disjointed shortest paths. More importantly, it can learn and adapt to media contents and changing network conditions while providing a multi-path protection and dynamic link recovery mechanism in SDN/NFV networks. The proposed MPTCP/SR SDN/NFV-based architecture consists of two layers or planes namely, the QoE-control and management plane and data plane.

[1]Experience Level Agreements (ELAs): Indicate a QoE-enabled counterpart to traditional QoS-based Service Level Agreements (SLAs) that convey the performance of the service in terms of QoE. ELAs establish a common understanding of an end-user's experience on the quality levels while using the service.

A. DATA PLANE

The data plane is the forwarding layer consisting of software/hardware devices such as Open vSwitch (OVS) that are interconnected using wired/virtual connections or wireless radio channels. The SDN switches in the data plane support SR technology. The OpenFlow [23]) protocol enables communications between the data plane and the QoE-control and management plane. The SDN controller communicates with the business applications running in the SDN/NFV platform through the Northbound Interface (NBI). Such exchange of information can be done using the RESTful API.

B. QOE CONTROL AND MANAGEMENT PLANE

The QoE control and management plane consists of an SDN controller which integrates the following modules that we extended in the POX controller in our implementations: the MPTCP module, QoE-management and the TE-Segment routing module. The other modules include a network information collector and a database module.

1) Network Information Collector (NIC)

The NIC collects network statistics, QoE requirements and system events especially when a link/node fails. It reports these network information (e.g., running status) to the SDN controller for an immediate action to be taken (e.g., recovery of a failed link).

2) MPTCP-based Flow Manager

The computation of shortest paths and allocation of these paths to an MPTCP connection are performed by the MPTCP-based flow module/manager. The MPTCP module uses admission control for multimedia services delivery to minimize the influence of network link congestion and improve the end user's QoE. That way, the MPTCP flow can be admissible on each link, if the sum of the rates of the allocated subflows does not exceed the link capacity (l_c). It is worth noting that, the number of the generated MPTCP subflows at the ingress switch source node is dynamically controlled by the MPTCP module. Moreover, the MPTCP module communicates with the QoE management module periodically such that resources can be assigned to subflow paths and meet their QoE requirements [29]. The SR

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

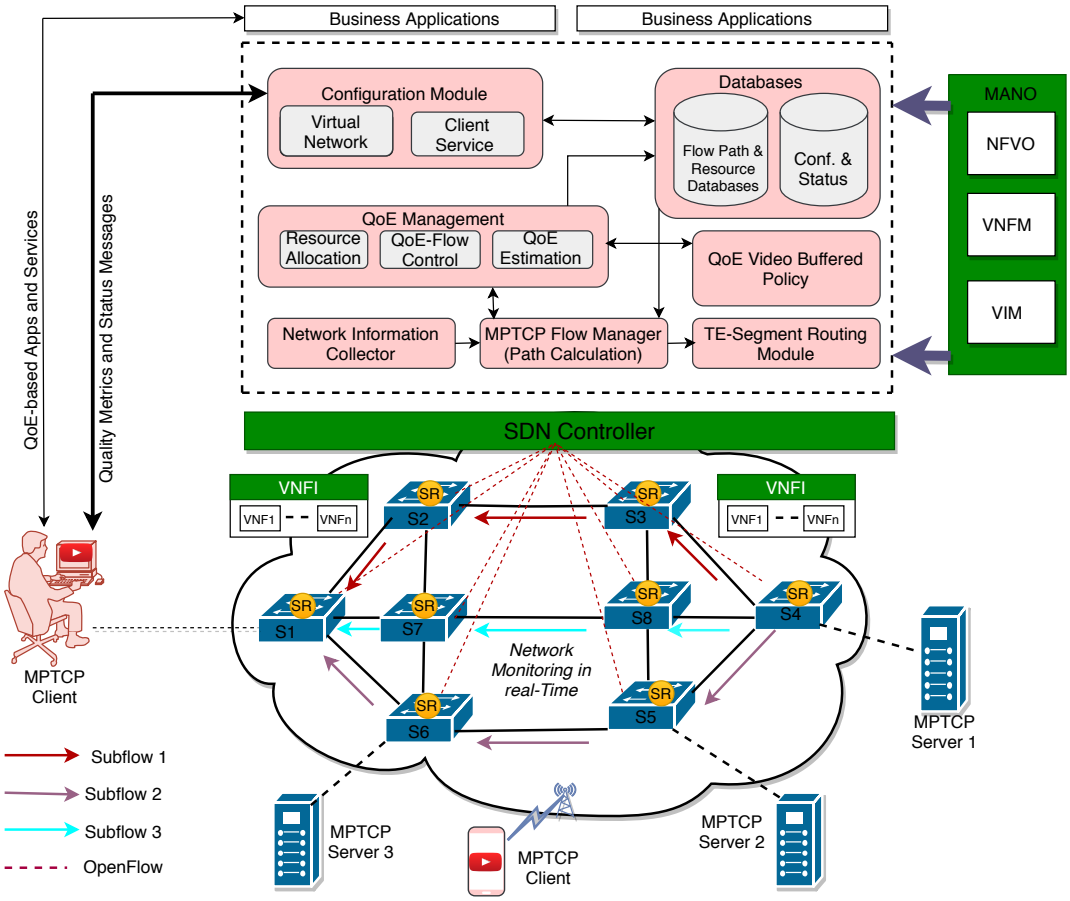


FIGURE 2: The Proposed QoE-aware, MPTCP SDN/NFV-based SR (QoEMultiSDN) Architecture.

technology implemented on SDN switches configures the forwarding tables with an ordered list of segments [28].

3) TE-Segment Routing Module

The TE-Segment Routing (TE-SR) module is responsible for mapping the computed subflow paths to SR paths. To achieve this, we extended the POX controller by introducing the SR-module to enforce MPTCP subflow paths through a minimal-length of SR paths on a hop-by-hop basis. As mentioned earlier, the SR paths represent the shortest list of SIDs where packets are routed to their destination points. As illustrated in Fig 2, suppose an MPTCP-based DASH client requests a video flow f whose ingress node is OVS $S4$ and the egress node is OVS $S1$. The path for MPTCP subflow $sf1$ with intermediate nodes $\{S_2,..S_{n-1}\}$ is: P_{sf1}={S1→S2→S3→S4}. The TE-SR module implemented in the controller considers the network topology graph and path P_{sf1} as inputs. It returns the list of segment labels after mapping the MPTCP subflows to path P_{sf1}. As an example to Fig 2, the ingress switch node $S1$ of the subflow path and the egress switch $S4$ are first considered. One of the assumptions made is that all network links can have different costs measured in terms of bandwidth, packet loss or delay values. The egress node is used as the SID node if only one shortest path exists and it equals to the MPTCP subflow path from $S1$ to $S4$. In that case, the mapping of this MPTCP subflow path to SR path ends. When a link failure occurs or it happens that the shortest path to be mapped is not equal to the subflow path or there exists more than one equal-cost shortest path, then other node segments can be considered [29], [21].

4) QoE-Management Module

The QoE management module consists of three sub-modules namely, the QoE-flow control, QoE estimation, and resource allocation. The QoE-flow control module performs video quality optimization and monitoring (network and user feedback) of video streaming services to enhance the end users' QoE [29], [21]. It also provides QoE-driven network management and service management as well as quality adaptation performance on the MPTCP DASH clients side. The QoE estimation performs measurements of Key Performance Quality or Indicators (KPQ/Is) for QoE metrics (e.g., stalling, reception quality [48]. It also determines and measures the network metrics such as system throughput, failure localization time, success rate, failure recovery time, packet loss and jitter. The description and the importance of these parameters in measuring the performance of the proposed architecture is given in section VI. The resource allocation module is

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

responsible for allocating resources to MPTCP flows based on the feedback received from the video MPTCP DASH clients. Dynamic allocation of resources is required due to the fact that, service providers are not either needed to control resource allocations in the Content Delivery Networks (CDNs) or monitor directly the status of the available system resources.

C. SDN CONTROLLER

The SDN controller is implemented as an extension to POX controller mainly to calculate MPTCP subflow shortest paths and maps them into SR paths as described in the above subsections. The SDN controller employs the list of segment labels to allocate and configure the SR path to the ingress SDN switch node. The SouthBound Interface (SBI), specifically using the OpenFlow protocol [37] message exchanges handles communication between the controller and network nodes in the data plane. These message exchanges are sent during a video flow request or system state change to reduce the overhead in OpenFlow switches and SDN controllers and control the network traffic [31]. The SDN controller is notified through a Packet-In Message when a heavy video flow is received at the switch. The Packet-Out Message enables the controller to manage the logical state of the switch, including its configuration and the details of flows.

It is worth mentioning that, the SDN controller is solely responsible for network failure detection through Operations Administration, and Maintenance (OAM) tools. SDN uses the OAM tools like Link Layer Discovery Protocol (LLDP) in layer 2 and layer 3 to provide tests for continuity check, connectivity verification, link trace, delay and loss ratio and detect link/node failure. For example, with POX controller, the failure detection is conducted using the POX Topology Discovery module [49]. When a link associated with the Openflow switches is detected as having been removed or failed, a *LinkEvent* message is raised by the Topology Discovery module. The *openflow.discovery* component in POX controller sends LLDP messages out of OpenFlow switches so that it can discover the network topology. For our proposed multipath protection and link failure recovery Algorithm 2 (to be discussed in Section V.D) that we propose in this paper, the SDN controller is implemented as an extension to POX and OpenDaylight controllers.

1) Database Module

The network monitoring status, reports and various configuration parameters are stored in the database. The SR subflow paths that have been used with their QoE requirements are stored and maintained in the database. When an MPTCP DASH client makes a new request from the video server, the MPTCP module finds in the database if there exists a transmission path that corresponds to the particular MPTCP connection. The video subflow is allocated to a specific path of a previously assigned subflow's path of the same MPTCP connection if the path is found in the database, otherwise, the MPTCP module computes a new path for the subflow. The TE-SR module then performs mapping of the new path to SR paths. The new path is stored and maintained in the database such that any incoming video subflow requests of the same MPTCP connection from the MPTCP DASH clients can use it.

2) Failure Recovery Handler Module

This module is responsible for detecting link/node failures in softwarized networks. It also provides multipath protections and dynamic link recovery to ensure that any failed parts in softwarized network are configured within a short period of time. It is important to mention that any changes of users and the running status of the SDN/NFV system can be reported immediately to the SDN controller [29] using the failure recovery handler. To summarize, our proposal provides four innovative benefits: (1) to speed up network transmission, (2) increase network throughput, (3) reduce link failure recovery time and overhead to the controller and, (4) lastly, to improve user's QoE for multimedia services in SDN-based networks.

3) NFV Management and Orchestration (NFV MANO)

According to the ETSI [50], network functions virtualization management and orchestration (MANO) is responsible for managing and orchestrating virtualized network functions (VNFs). It consists of the NFV Orchestrator (NFVO), VNF Manager (VNFM) and the Virtualized Infrastructure Manager (VIM). The NFVO performs orchestration and lifecycle management of physical and software resources that support infrastructure virtualization. The NFVO also performs global resource management, network services (vCompute, vStorage and vNetwork resources) within one operator's infrastructure domain. The proposed QoEMultiSDN can also provide an efficient orchestration, QoE control and management of future multimedia services via instantiation, validation and authorization of NFVI resources requests. The VNFM is responsible for lifecycle management of VNF instances, overall coordination and adaptation role for configuration and event reporting between NFVI. A single or multiple VNF instances of the same or different types can be managed by a VNFM. VIMs control and manage NFVI physical and virtual resources.

V. QOE-DRIVEN MULTIPATH SOURCE ROUTING AND DYNAMIC LINK RECOVERY ALGORITHM

The multi-flow strategy is the core of our proposed architecture. It enables us to assign video flows to the specific path while taking into account parameters involved in the path and the flow requirements such as available bandwidth, packet loss and delay of the network link. To achieve this goal, the proposed architecture in Fig. 2 performs continuous monitoring of the network parameters and adapt to the network resources to satisfy the multimedia flow necessities. We formulate our mathematical model based on two well known problems that have been addressed by operations research namely: *the Multi-Commodity Flow Problem* and the *Constrained Shortest Path Problem*. Table 2 shows a

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

summary of notations used in the problem analysis and the formulated algorithm.

A. MULTI-COMMODITY FLOW PROBLEM

The multi-commodity flow problem (MCFP) deals with the assignment of commodity flows from source to destination in a network. MCFPs are highly relevant in several fields including transportations and telecommunications. MCFP is a particular network flow problem where multiple commodities (e.g., different multimedia flow demands/services), should be sent from various sources to distinct destination nodes. This way, we assume to have a network of interconnected nodes where each link is associated with a particular cost (e.g., bandwidth). Moreover, the traffic along the links between each pair of nodes consumes an amount of bandwidth during video streaming. Our goal is to find the optimal set of routes through the network for each of those commodities with the minimum total flow cost. The constraint is that the total flow on a link should not exceed the link capacity. To solve MCFPs, two necessary constraints must be considered. The first is the travel demand, which means that all the commodities need to be transported to their destinations. The second is the edge capacity constraint. This means that the flow on each edge can not exceed flow capacity.

A directed graph G = (V,E) that represents a softwarized network is considered. E indicates a set of edges in G and V is the set of network nodes such as OVS. We consider that the link cost $W(e)$ in the network is defined as a non-negative integer on each edge $e \in E$. It is worth noting that, the link cost in real-world network operation can be represented by QoS parameters such as packet loss, bandwidth, link utilization or delay. We denote h as multiple shortest paths required for subflow transmissions from source to destination in softwarized network. These paths are computed based on the cost of each link. Each pair connecting SDN switches in the data plane consist of a link bandwidth denoted as b_w. During video transmission, we suppose that a subflow sf of an MPTCP connection k requires a bandwidth of B^k_{sf}. Moreover, we denote $p(s, t)$ as the shortest path of a traffic-flow demand f that originates from the source node $s \in V$ to its final destination $t \in V$ in softwarized network. The constrained shortest path and the multi-flow commodity models [51] are the well-known strategies considered in this paper to achieve an efficient routing of multiple video subflows in softwarized network. Subject to the link capacity and other constraints (packet loss, delay etc.), the main objective of these models is to perform the transmission of MPTCP subflows using a path with a minimum link cost/weight. The link cost, We_{ij} is defined using equation 1 as the sum of packet loss (pl_{ij}) and delay (dl_{ij}) values. The scale factors α, and β which are ≥ 0 are multiplied to these values such that:

$$We_{ij} = \alpha \times dl_{ij} + \beta \times pl_{ij}, \forall (i,j) \in E. \tag{1}$$

To route the MPTCP subflow paths with a minimum cost, we define an objective function shown in equation 2. For each link (i, j) of all shortest paths selected for transmission of an MPTCP subflows of connection k, the objective function that minimizes cost while achieving an optimized QoE for the end-users is given by

$$O_f = min \sum_{(i,j)\in E} \sum_{k\in K} (We_{ij} sf^k_{ij}) \tag{2}$$

where sf^k_{ij} indicates the subflow of an MPTCP connection k routed on the link (i,j). The following constraints for the delay, packet loss and bandwidth are defined to ensure that they can not exceed the maximum acceptable values.

B. CONSTRAINED SHORTEST PATH MODEL

The Constrained Shortest Path problem (CPS) is an extension of the Shortest Path (SP) problem that calculates the shortest path while fulfilling a set of specific system constraints. In particular, we have a network of nodes where each link is defined by various parameters, e.g., in our case the delay and the packet loss. Moreover, each link has a cost associated with it. The goal is to find the optimal path between a source and destination with the minimum cost after pruning those links that violate a given set of constraints. We consider different constraints for the delay, packet loss and bandwidth to ensure that they can not exceed the maximum acceptable values. The first constraint defined in equation 3 is related to the flow balancing while equations 4 and 5 refer to the maximum acceptable value for the packet loss and the delay respectively. Equation 5 is the available bandwidth for each link as bounded by an arc capacity constraint.

Constraint 1: We adopt in this paper, the well-known flow conservation law as shown in equation 3 to guarantee the subflow balancing in an SDN/NFV system and make sure that, the total flow incoming into each vertex, V is equal to the total flow outgoing from the same vertex, with the exception of the MPTCP client and MPTCP server.

$$\sum_{(i,j)\in E} sf^k_{ij} - \sum_{(j,i)\in E} sf^k_{ji} = \begin{cases} sf^k & \text{if } i{=}s_k \\ -sf^k & \text{if } i{=}t_k \\ 0 & \text{if } i \neq s_k, t_k \end{cases} \quad \forall i \in E, \forall k \in K \tag{3}$$

Constraint 2: The maximum acceptable delay D^k_{max} and packet loss D^k_{max} impose the limit for each subflow of an MPTCP connection k.

$$\sum_{(i,j)\in E} (pl_{ij} sf^k_{ij}) \leq P^k_{max}, \forall k \in K \tag{4}$$

$$\sum_{(i,j)\in E} (dl_{ij} sf^k_{ij}) \leq D^k_{max}, \forall k \in K \tag{5}$$

Constraint 3: The available bandwidth for each link is bounded by an arc capacity constraint as defined in equation 6.

$$\sum_{(i,j)\in E} B^k_{sf} sf^k_{ij} \leq b_{ij}, \forall k \in K \tag{6}$$

We define the above parameters that are used to formulate the objective function and the proposed system constraints.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

TABLE 2: SUMMARY OF NOTATIONS IN PROBLEM ANALYSIS

Symbol	Meaning
b_{ij}	Available bandwidth on link (i, j)
We_{ij}	Cost of link (i, j) calculated using equation 1
dl_{ij}	Delay on link (i, j)
$\hat{B}(i)$	The estimated available bandwidth that can be used to determine the bitrate level for the next chunk i to be downloaded
pl_{ij}	Packet loss on link (i, j)
P^k_{max}	Maximum acceptable packet loss of a subflow of an MPTCP connection k
D^k_{max}	Maximum acceptable delay of an MPTCP subflow of connection k
B^k_{sf}	The required bandwidth of an MPTCP subflow sf of connection k
sf_k	Amount of subflow of an MPTCP connection k to be sent from $s \in$V to $t \in$V
s_k	The source of a subflow of an MPTCP connection k
t_k	The destination of a subflow of an MPTCP connection k
$\alpha \geq 0$	Scale factor for the delay
$\beta \geq 0$	Scale factor for the packet loss

C. QOE-DRIVEN MULTIPATH ROUTING AND QUALITY OPTIMIZATION ALGORITHM IN SOFTWARIZED NETWORKS

The QoE-driven multipath routing and quality optimization algorithm is implemented on an SDN source routing platform using Mininet and the POX controller. To formulate a QoE-driven quality optimization algorithm, an assumption is made that, during video streaming, a total of N chunks each lasting for t_s seconds are available in one video sequence for download. Based on the available bandwidth, the MPTCP DASH client requests video segments that satisfy their quality levels. The set of quality levels requested are, Q_L = {q_1, q_2, . . . , q_N}. The chain of bandwidths to download each video segment are, B_C = {b_1, b_2, . . . , b_N}. The MPTCP client can request an average video quality $V_{avg}(x, y)$ defined by equation 7.

$$V_{avg}(x, y) = \frac{1}{N} \sum_{1=x}^{N} q_{x,y} \tag{7}$$

For QoE evaluation using the proposed QoE-based multipath routing and quality optimization algorithm, the initial delay, buffering events, stall duration and the bitrate level variation are considered. Before the initial delay I_{delay} of the video segment x is requested, then the initial delay of $I_{delay}(x-1)$ is accumulated during video streaming. Then, if the bitrate level y is selected for x, the initial delay will increase by $\frac{S_{x,y}}{\hat{B}(x)}$. From this formulation, the impact of the initial delay as defined in [52] can be given by equation 8.

$$I_{delay}(x, y) = 3.2I_{delay}(x-1) + \frac{S_{x,y}}{\hat{B}(x)},\ Bf_{size} < Bf_{min} \tag{8}$$

Bf_{size} is the buffer size while Bf_{min} is the buffer threshold such that: (a) if the MPTCP client is in buffering state, then Bf_{size} < Bf_{min}, (b) and $Bf_{min} \leqslant Bf_{size}$ < Bf_{max}-S_d when the client is in playout state where S_d is the duration (e.g. 2, 4, 6, 8 etc) of video segments in seconds.

To determine the impairment factors on video quality due to stalls, the combination of a number of stalls and stall duration are considered. Note that, when the MPTCP client is rebuffering, the requested video segment x will introduce a stall duration by $\frac{S_{x,y}}{\hat{B}(x)}$ whereas the buffer size is Bf_{size} + S_d - $\frac{S_{x,y}}{\hat{B}(x)}$ when the segment x is downloaded. A threshold T_{ST} for stalling events that avoid buffer length can be defined as:

$$T_{ST} = Bf^{'}_{size} + \frac{S_{x,y}}{\hat{B}(x)} - S_d \tag{9}$$

$Bf^{'}_{size}$ is the warning threshold defined by γ Bf_{min} + (1-γ) Bf_{max} that shows the possible occurrence of buffering events. The number of stalls $N_{stall}(x, y)$ and stall durations $S_{drtn}(x, y)$ for the next video segment x with bitrate level y are defined based on [53].

The impairment indicators for stalls $I_{stall}(x, y)$ as defined in [53] can be calculated using a combination of $I_{delay}(x-1)$ and $S_{drtn}(x, y)$ as shown in equation 10.

$$\mathrm{I}_{stall}(x, y) = (3.8)S_{drtn}(x, y) + 4.2N_{stall}(x, y) - 2.6\sqrt{N_{stall}(x, y)S_{drtn}(x, y)} \tag{10}$$

The bitrate level variation $B_{LV}(x, y)$ is another parameter which affects the end-user's QoE because of two factors, namely, the video quality and the switching impact defined as $V_{Quality}(x, y)$ and $S_{impact}(x, y)$. The $V_{Quality}(x, y)$ is calculated by equation 11. 3.8 and 4.2 are constants as defined in [53].

$$V_{Quality}(x, y) = \sum_{x=1}^{N} VQM_{l(x)} e^{\frac{0.02S_{drtn}S_d}{N}} \tag{11}$$

The $V_{Quality}(x, y)$ is the weighted mean of the $VQM_{l(x)}$ of N chunks where $e^{(0.02S_{drtn}S_d)}$ is the weight of each video segments.

The $S_{impact}(x, y)$ is defined using the following equation.

$$\mathrm{S}_{impact}(x, y) = \frac{1}{N} \sum_{x=1}^{N} [VQM_x - VQM_{x+1})^2) sign(VQM_{x+1} - VQM_x) \tag{12}$$

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

where

$$sign(x) = \begin{cases} 1 \text{ if } x > 0 \\ 0 \text{ otherwise} \end{cases} \quad (13)$$

The bitrate level variation $B_{LV}(x, y)$ is calculated as the sum of $V_{Quality}(x, y)$ and $S_{impact}(x, y)$ such that

$$B_{LV}(x, y) = B_1 \times V_{Quality}(x, y) + B_2 \times S_{impact}(x, y) \quad (14)$$

where B_1 and B_2 are the coefficients with values of 75.6 and 48.2 respectively [53].

To balance the network load and avoid a bottleneck link during transmission of video services in softwarized networks, a *link criticality* l_c parameter is introduced. We consider in this paper P_{st} to be a set of the first h computed shortest paths from an MPTCP server to MPTCP client. We also assume that P_{st}(c) is the number of times that link $e \in E$ is included in the first h computed shortest paths. For any pair of communicating nodes in softwarized network, the occurrence rate of link e in the first h shortest paths can be calculated as P_{st}(c)/h. We are then able to calculate the load expected on link e which can be given by the sum of the number of traffic-demands on that link collected from all possible paths that connect the transmission from source and a destination node in sofwarized networks. In that respect, the link criticality l_c is defined and calculated using equation 15, where TD is the traffic demand recorded and stored in the database after every time T.

$$l_c = \sum_{(s,t) \in TD} P_{st}(c)/h \quad (15)$$

The l_c that avoids a bottleneck link and balance the load in softwarized network is updated by the controller after every time T. When the computation of multiple shortest paths is completed based on the described link constraints, the SDN controller then performs the mapping of MPTCP subflows paths into SR paths. The QoE metrics are reported in real-time during video transmission from the video server to the MPTCP DASH client. We summarize the implementation steps of the proposed QoE-aware MPTCP/SR SDN/NFV-based approach using Algorithm 1. The required TCP throughput for a specific video resolution of [1080p, 720p, 480p, 360p, 240p] which is selected by Algorithm 1 when an MPTCP server receives a request from an MPTCP client is [5 Mbps, 2.5 Mbps, 1 Mbps, 0.725 Mbps, 0.325 Mbps] respectively.

Fig. 4 shows a workflow of QoEMultiSDN which consists of Algorithm 1 and 2. Algorithm 1 indicates the QoE-driven multipath and quality optimization strategy while Algorithm 2 is the multipath protection and dynamic recovery of link failures in SDN/NFV networks.

Algorithm 1: QoE-driven Multipath and Quality Optimization Algorithm

input : flow f, *N_topology* $G = (V,E)$

Compute link cost based on equation 1, bandwidth;
Find all subflow shortest paths $p \in$*P in the network*;
foreach $p \in$*Psrc*$\rightarrow$*dst* **do** ,;
if *p.used* $+sf_k.B^k_{sf} < b_{ij}$ **then return** p ;
else Go to step 3;
Perform mapping of subflow paths p_{sf} *into SR paths*;
output: List of Segment labels *SL*
Save path p with its associated list of *SL* in DB_p;
Start MPTCP subflow transmission based on client's QoE requirements;
Go to **Algorithm 2** when link(i,j) fails
Calculate buffer size Bf_{size} and $\hat{B}(x)$
Calculate $B_{LV}(x, y)$, $V_{Quality}(x, y)$ and $S_{impact}(x, y)$ using 14, 11 and 12
if $i < Bf_{min}/ S_d$ **then** ;
Find $I_{delay}(x, y)$ using 8;
else $I_{delay}(x, y) \leftarrow I_{delay}(x - 1, j)$;
if $Bf_{max} < Bf_{min}$ **then** ;
Find $N_{stall}(x, y)$, $S_{drtn}(x, y)$;
else Go to step 9;
end if;
end if;
end for;
if f_{new} *is a new flow* **then** Query DB_p to find the paths for subflows of f_{new};
if pf_{new} *is not in* DB_p **then** Go to step 2;
Issue path adding request for f_{new} **end if**;
end if;
Continue transmission as long as $B^k_{sf} < b_{ij}$;
Use the l_c and congestion index to avoid congestion

D. MULTIPATH PROTECTION AND DYNAMIC LINK-FAILURE FREE ALGORITHM IN SDN/NFV

When a link failure is detected in traditional networks, crankback routing [46] is normally applied where the traffic is recursively returned to the source node from which the traffic can be forwarded to the destination node again. However, the crankback routing increases network recovery time because the failure detection window also increases due to the longer path lengths. We provide in this paper new approaches for path protection and link recovery mechanisms using SR as shown in Fig. 3. Upon a link failure, the node where the link failure is detected (step 1) reroutes the data packet until the convenient route is found where the packets can be forwarded to their destination point. The novelty of this approach is that, the same data packets of MPTCP flow are tagged first (e.g. with a SR label which contains information on the failed link) and then sent back through the primary path. After receiving the tagged packet (step 2), a reroute node (e.g., $S2$) can then respond to the link/node failure and forward the tagged packets (step 3) to their destination node. When the reroute node processes the first tagged packet, a state transition is performed in the OpenFlow switch. A failover table is created and install

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

backup actions where all subsequent packets coming from the source node are forwarded on the reroute node. It is worth mentioning that, when link failures occur in the network, segments can be repaired such that new backup paths are computed for each link in the primary path as shown in Fig. 3. The benefits of this approach are the reduction of path costs, network recovery time and backup path length. To achieve

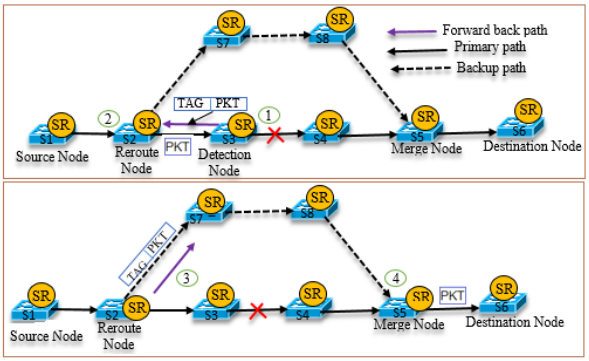


FIGURE 3: An Example of Path Protection and Link Recovery Mechanisms

fast recovery after a link failure, the failed link l_{ij} has to be bypassed using the computed local reroute paths such that:

$$X_{s_f,t_f}^{p_i} = 0, \forall p \in N \tag{16}$$

Where N is the set of new computed shortest paths after the link (i, j) has failed and p_i is the ith reroute path, for $1 \leq i \leq n$. n is the number of new computed reroute paths when (i, j) fails. The recomputed paths should also have a cycle avoidance constraint defined as

$$\sum_{t \in V,(s,t) \in E} (X_{s,t}^{p_i}) \leq 1, \forall s \in V \tag{17}$$

When the failure occurs in the SDN/NFV network, the bandwidth of the disrupted MPTCP flow assigned to the rerouting path p_i can not exceed the available capacity of path p_i.

When link (i,j) fails, then it is pruned using a script shown in Algorithm 3 and the network topology is updated. The algorithm takes the updated network topology as input and recomputes the link cost using equation 1. The SDN controller calculates the shortest path where MPTCP flows should be transported from s to t. It then performs the mapping of MPTCP subflow paths p_{sf} into SR paths. When the client and server establish an MPTCP connection using a three-way handshake, the transmission of subflow is initiated based on the user's service requirements. The controller monitors system events and status of network topology. The overall procedure of our proposed approach is summarized in Algorithm 2. The novelty of this approach is that it does not need crankback routing as in traditional routing mechanisms and the failure detection window and recovery time are minimized.

Algorithm 2: Path Protection and Recovery of Link Failure (PathReLief)

1 *Remove failed link(i,j) by using* ***Algorithm 3*** ;
input : *New_topology* $G^{'} = (V^{'}, E^{'})$
2 *Find all subflow shortest paths* $p^{'} \in P^{'}$ *in the network*;
3 **foreach** $p^{'} \in P^{'}$ **do** ,;
4 **if** $p^{'}.used + sf_k.B_{sf}^{k} < b_{ij}$ **then return** $p^{'}$;
5 **else** Go to step 3;
6 **end if**;
7 Map all subflow shortest paths $p_{sf}^{'}$ into $SR^{'}$ paths
output: List of Segment labels $SL^{'}$
8 Go to step 6 in **Algorithm 1 end for**

Algorithm 3: A Script to Remove a Link from the Network Topology

1 *def removeLink(self, node1, node2, port1=None, port2=None, **opts)*;
2 **if** *not opts and self.lopts:* **then** ;
3 self.addPort(node1, node2, port1, port2);
4 *key = tuple(self.sorted([node1, node2]))*;
5 *self.link_info[key] = opts*;
6 *self.g.remove_edge(*key)*;
7 return key

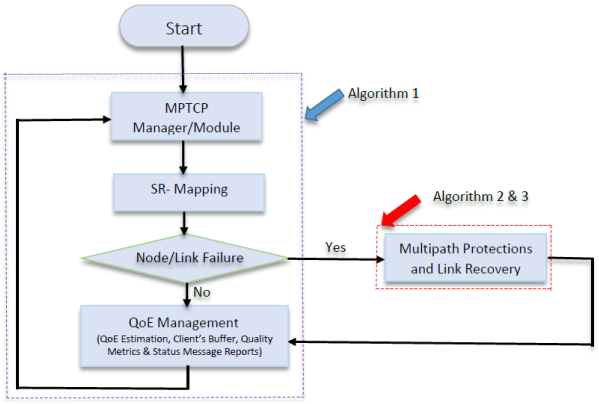


FIGURE 4: The Workflow of the QoEMultiSDN Scheme

VI. SIMULATION SETUP & PERFORMANCE ANALYSIS AND EVALUATION

A. EXPERIMENTAL SETUP

We use the experimental testbed shown in Fig. 5 to demonstrate the credibility and feasibility of QoEMultiSDN. We installed the MPTCP v0.92 in two Virtual Machines (VMs) running the Linux Ubuntu V16.04 LTS. We then installed the Mininet V2.2.2 in one VM and used it to develop a fat tree network topology shown in Fig. 5. Our network topology consists of redundant links connecting OpenFlow switches at the edge, aggregation and core layers. The total number of

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

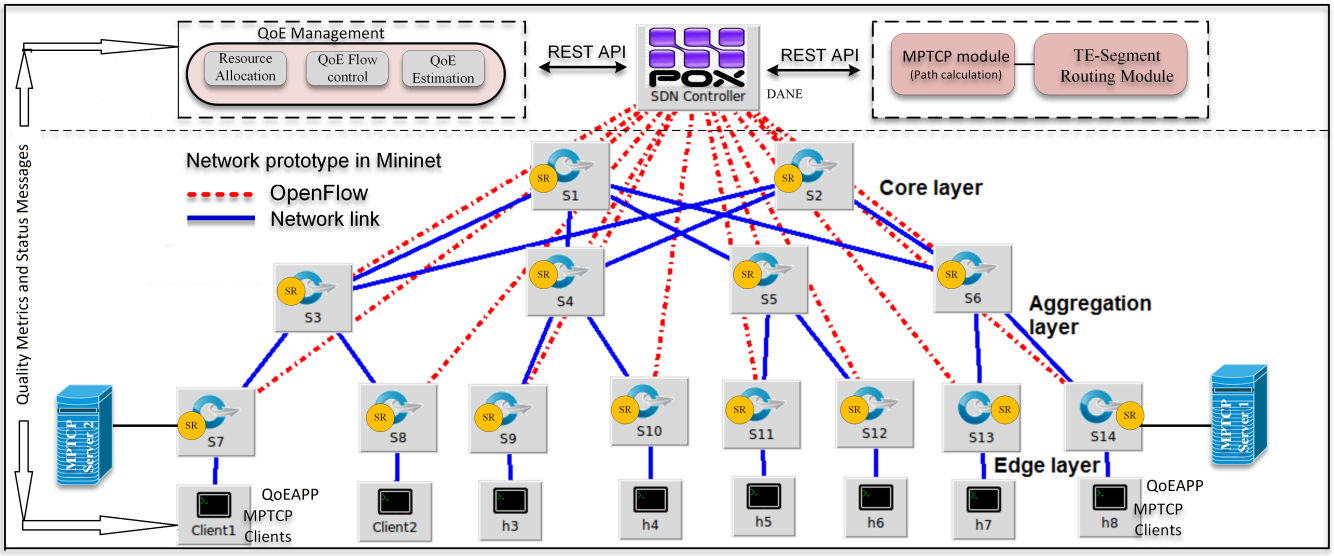


FIGURE 5: Experimental Testbed

SDN switches at the edge, aggregation and core layers are 8, 4 and 2 respectively. The SDN controller was extended by implementing the MPTCP-flow manager and the SR module in the POX controller running in the second VM. Apache Server was installed on two MPTCP server machines and attached them on the mininet network as shown in Fig. 5. We make sure that the client and server sides in our testbed support both TCP and MPTCP. The VLC-DASH plugin [54] was used as a DASH client. We consider both user and network KPIs to evaluate our proposal. The former include QoE-related metrics which are reported by the QoEAPP as shown in Fig. 5, while the latter focuses on the overall network performance metrics. QoEAPP is implemented in Algorithm 1 following the principles of Server and Network Assisted DASH (SAND) [55] architecture such that DASH-Aware Network Elements (DANE) and DASH clients can communicate through quality and status messages. Status messages are sent from DASH clients to DANE while quality metrics are messages sent from the DASH clients to video servers. Using SAND approach, our implementation gives more flexibility in deploying the QoEMultiSDN such that, we can collect DASH client's QoE information (video quality switches, startup delay etc).

We designed two experimental use cases: use case 1 to demonstrate video quality measurements. Use case 2 provides a demonstration of failure recovery, localization time and link capacity occupation in future softwarized network. We compared the performance of QoEMultiSDN, MPTCP and the regular TCP using the following network and video delivery quality metrics: (a) quality switches, (b) startup delays, (c) system throughput, (d) success rate, (e) video streaming reception quality, (f) failure recovery time, (g) failure localization time and (h) the link capacity occupation.

TABLE 3: List of used parameters

N.Layer	Bandwidth (Mbps)	Delay (ms)	Packet Loss (%)
Edge	3	15	1
Aggregation	2	20	3
Core	3	30	3

B. USE CASE 1 - VIDEO QUALITY MEASUREMENTS

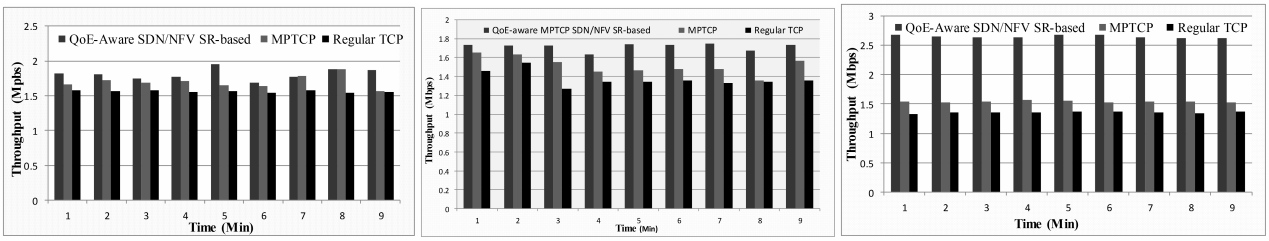
This experimental design investigates the video quality measurements in terms of quality switching, startup delays, network throughput and success rate during video streaming.

1) Comparison of Video Quality Switches and Startup Delays

We conducted two experiments in use case 1. We collected the number of quality switches, startup delays and the success rate in the first experiment as shown in Table 4. We compared the performance of our proposal and the regular TCP after performing a video request from the MPTCP server three times using a 6 sec segment length. We used a 6 sec because it provided a good encoding efficiency and flexibility for stream adaption to bandwidth changes compared to others (4 sec, 8 sec and 10 sec). The success rate is defined as the percentage of segments at maximum bitrate over all segments used during a video streaming session. The quality switches define how many times the video quality changes from one bitrate to another during video streaming because of network bandwidth variations. It is worth noting that, few video quality switches and low startup delays lead to a good user's video watching experience. As shown in Table 4, in all three experiments, our QoE-aware MPTCP/SR SDN-based proposal performs better compared to the regular TCP. Note that, for video quality switches, startup delays and success rate, we only compare our approach in this experiment with the regular TCP.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access



(a) System Throughput for 360p (b) System Throughput for 480p (c) System Throughput for 720p

FIGURE 6: Comparison of System Throughput for QoE-Aware MPTCP SDN/NFV-based SR, MPTCP and Regular TCP

TABLE 4: Selected QoE comparison results for Regular TCP and QoE-Aware MPTCP SDN/NFV-based SR

	Regular TCP			QoE-Aware MPTCP SDN-based SR		
Experiment	Quality Switches	Startup Delays (sec)	Success Rate (%)	Quality Switches	Startup Delays (sec)	Success Rate (%)
1	8	6.60	85	2	2.21	96.64
2	7	6.30	87	3	1.97	97.80
3	5	5.60	84	1	1.60	97.01

In the second experiment, we further compared the performance of regular TCP, MPTCP and QoEMultiSDN (MPTCP/SR SDN/NFV-based) approach using network throughput and the end user's QoE. We use the same "Big Buck Bunny" video in this experiment which was encoded using the H.265 or High Efficiency Video Coding (HEVC). The video encoding rate of 2.496 Mbps, 1.536 Mbps and 1.0888 Mbps were used for 3 different resolutions (720p, 480p and 360p) respectively. The network QoS parameters shown in Table 3 were configured on the links of network topology. The default configurations of MPTCP V0.92 was enabled in two VMs, same to that of experiment 1. The MPTCP path manager was set to a *full-mesh* to limit each MPTCP connection to have 3 MPTCP subflows for transmission. We perform 40 runs of video streaming sessions from the MPTCP server to an MPTCP DASH client. The DASH client for this experiment was also based on the VLC-DASH plugin [54] which reports the performance of system throughput and QoE-related metrics such as the video streaming reception quality. The system throughput is computed from Payload_bits/download_time; where a *Payload_bits* is the number of extracted bits of the video content per single unit time.

2) System Throughput

Fig. 6a indicates the system throughput using a video resolution of 360p for QoEMultiSDN, MPTCP and the regular TCP. Fig. 6b and Fig. 6c show the comparison results for a video resolution of 480p and 720 respectively. We note that the performance of throughput for our proposed strategy is better compared to MPTCP and the regular TCP. This is so because the QoEMultiSDN or the QoE-aware MPTCP SDN/NFV SR-based approach exploits multipath transmission using shortest paths and employs SR which does not require any path signaling in the data plane. Table 5 indicates the average throughput achieved by QoEMultiSDN (QoE-aware SDN/NFV SR-based), MPTCP and TCP for 360p, 480p and 720p after 40 runs of video streaming. As expected, the average system throughput of QoEMultiSDN for all video resolutions is higher compared to MPTCP and the regular TCP.

TABLE 5: Average Throughput in Mbps

Video Resolution	QoEMultiSDN	MPTCP	TCP
360p	1.87	1.73	1.55
480p	1.76	1.61	1.36
720p	2.66	1.56	1.37

3) Video Reception Quality Measurements

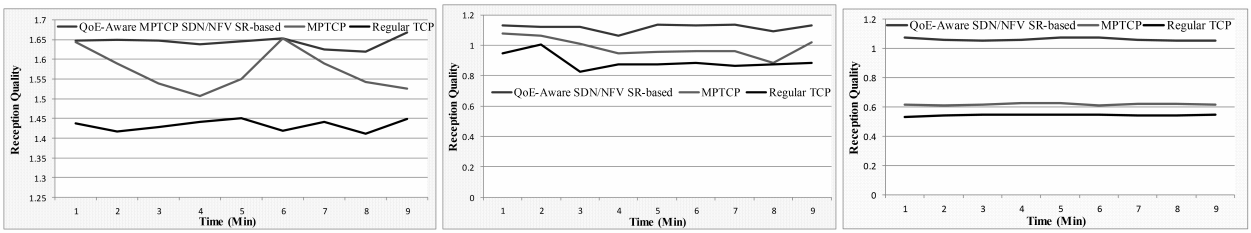
The reception quality ρ [48] metric is used to measure the video quality delivered to DASH clients. (ρ) is defined as the ratio between download throughput and video encoding rate. The delivered video to the DASH client has a good reception quality if $\rho > 1$ otherwise, the video delivered has poor quality. The video reception quality for a video resolution of 360p and 480p is shown in Fig. 7a and 7b respectively. Fig 7c indicates the reception quality comparison results using a video resolution of 720p. For all video resolutions (360p, 480p, 720p), our proposal performs better compared to MPTCP and regular TCP. This is so because the multipath transmission and SR provides higher system throughput for QoEMultiSDN approach for the selected video resolutions. The ρ for a video resolution of 360p is higher because its encoding rate is low (1.0888 Mbps) compared to the ρ of 480p and 720p (i.e, 1.536 Mbps, 2.496 Mbps respectively).

4) QoE and its Metrics

To evaluate the QoE performance metric in terms of the Mean Opinion Score (MOS) we employed the QoE model proposed in [56] that takes into consideration four metrics

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access



(a) Reception quality for 360p (b) Reception quality for 480p (c) Reception quality for 720p

FIGURE 7: Comparison of Video Streaming Reception quality for QoE-Aware MPTCP SDN/NFV-based SR, MPTCP and Regular TCP

for calculating the MOS values. These metrics are selected bitrate, stall duration, startup delay, and bitrate switches. The calculation of QoE is given in [56] and the outputs of QoE values are in the range from 1 to 5. We also calculated the live latency which indicates the time taken from capturing to rendering during a live video streaming. The live latency values during a live video streaming session are provided by the VLC dash plugin [54] player and it includes the encoding time. Table 6 provides a summary of average performance metrics for QoEMultiSDN, MPTCP and regular TCP after 40 runs of video streaming sessions. We note that, QoEMultiSDN outperforms the MPTCP and TCP approaches for all video resolutions. For all bandwidth profiles, QoEMultiSDN selects the best bitrate levels compared to MPTCP and TCP techniques. The selected bitrate levels for QoEMultiSDN remains in the network capacity range (e.g., 0.89 Mbps, 1.3 Mbps), 2.92 Mbps) and maintains low latency while avoiding a larger number of video stalls and startup delays. Similar results are observed for video bitrate switches where QoEMultiSDN achieves low bitrate switches (2, 4, 3) on average for 360p, 480p and 720p respectively. This results into an overall higher end users QoE (4.2, 4.0, 4.1) for all video resolutions while MPTCP and the regular TCP achieves low viewer QoE because of suffering from frequent stalls, long stall durations, many bitrate switches and startup delays.

5G networks are making the difference in supporting multimedia streaming services by offering enhanced service availability and consistency, massive connections with high data rates and low latency. The reduced latency and high data rate offered by 5G combined with the MPTCP/SR technologies, imply that QoEMultiSDN will easily provide a high level of multimedia streaming services with guaranteed end-users service quality to customers from service providers.

C. USE CASE 2 - MEASUREMENTS OF FAILURE RECOVERY AND LOCALIZATION TIME IN SOFTWARIZED NETWORKS

We evaluated the performance of localization time and failure recovery time using the proposed Algorithm 2. To achieve this, we installed the POX and OpenDaylight controllers in the second VM. It is worth noting that the link between the pair of switches that we selected were failed using the pseudocode shown in Algorithm 3. For each pair of SDN switches, the failed link was repeated 20 times for both POX and OpenDaylight controllers. We then collected the performance results of failure recovery time and localization time based on the PathReLief algorithm. The failure localization is defined as the process used to identify the exact point where the failure of a link/node has occurred in SDN/NFV-based network. The Failure Localization Time (FLT) is the time that the SDN controller takes to get information about a failed link from the POX or OpenDaylight *Topology Discovery* module to the time it finds the exact failure location in the network. The Failure Recovery Time (FRT) is the time that the SDN controller takes to receive the information of the exact point where link/node failure has occurred and install the new flow failover tables in switches. We adopt the optical networks failure recovery model presented [57] to calculate the FRT. The main idea of implementing the path protection and link recovery algorithm 2 in POX and OpenDaylight controllers was to investigate how these controllers react to link/node failures in softwarized network.

1) Failure Recovery Time (FRT)

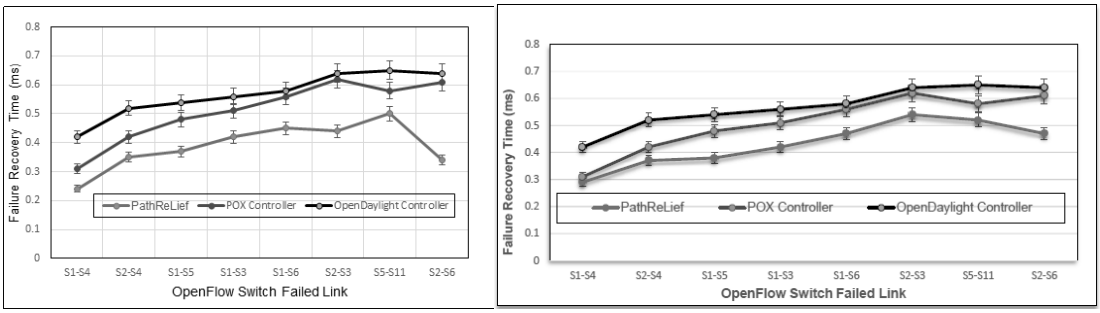
The performance comparison results of failure recovery time using the PathReLief algorithm in POX and OpenDaylight controllers are shown in Figs. 8a and 8b. The native topology discovery approaches used in POX and OpenDaylight controllers perform poorly compared to the performance of PathReLief algorithm. Not only PathReLief algorithm uses the MPTCP for transmitting the traffic flows but it also employs the SR technology on SDN switches. That way, it makes the transfer of network traffic from the failed node to the SDN controller. Note that, we disable the SR in the POX and OpenDaylight controllers when evaluating the performance of the failure recovery and localization time using the native topology discovery approaches in both controllers. The PathReLief algorithm takes only 0.24 *ms* when {S1→S4} fails to recover the failed link of the SDN/NFV-based network and install failover tables in switches. The link {S1→S4} connecting the switches belongs to the aggregation and core layer which are closely connected to the SDN controller. The direct connection of switch $S1$ to the SDN controller contributes greatly to minimize the time

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

TABLE 6: A Summary of Average Performance Metrics for QoEMultiSDN, MPTCP and Regular TCP

Technique	Avg. Live Latency (s)	Avg. No. of Stalls & (Duration) s	Avg. Selected Bitrate (Mbps)	Avg. Startup Delay (s)	Avg. No. Bitrate Switches	Avg. Success Rate (%)	Video Resolution	Avg. MOS
	1.2	4 & 8	0.89	2.21	2	95.40	360p	4.2
QoEMultiSDN	1.4	6 & 7	1.3	2.26	4	96	480p	4.0
	1.3	9 & 10	2.92	1.80	3	97.5	720p	4.1
	2.1	6 & 10	0.82	4.45	5	89	360p	3.4
MPTCP	3	9 & 11	1.1	4.67	6	87.3	480p	3.1
	2.5	13 & 16	2.71	3.4	4	90	720p	3.2
	4.2	7 & 14	0.74	6.6	8	85	360p	2.7
TCP	3.4	11 & 18	1.01	7.2	10	84.6	480p	2.4
	3.1	21 & 28	2.51	5.8	9	83.8	720p	2.2



(a) Failure Recovery Time using POX Controller (b) Failure Recovery Time using OpenDaylight

FIGURE 8: The Performance of Failure Recovery Time using POX and OpenDaylight Controllers

required to recover the failed link in the SDN network. When {S5→S11} fails, the PathReLief algorithm and the topology discovery modules implemented in the POX and OpenDaylight controllers takes (0.5 *ms*, 0.58 *ms*, 0.65 *ms*) to recover the failed link in the network. The failure recovery time is the largest because link {S5→S11} belongs to the edge layer and aggregation layer. However, the PathReLief algorithm takes shorter time compared to that of the POX and OpenDaylight controllers.

2) Failure Localization Time (FLT)

The comparison results for FLT are shown in Fig. 9a. The performance of the native approaches used in POX and OpenDaylight controllers are poor compared to the PathReLief algorithm. The failure localization time for PathReLief, POX and OpenDaylight are 48 μs, 81 μs and 94 μs respectively when {S1→S4} fails. As indicated in the fat-tree topology in Fig. 5 the time taken for the controller to detect the exact point of a failed link is shorter for PathReLief algorithm because link {S1→S4} connects the core and the aggregation layer. The localization time is 78 μs, 85 μs and 96 μs for PathReLief, POX and OpenDaylight controllers respectively. This is so because link {S5→S11} connects the aggregation layer and edge layer. This makes the backup path length to increase in the SDN/NFV-based network. Note that, MPTCP and SR also would enable the network to transmit the network traffic fast in the SDN/NFV system.

3) Link Capacity Occupation (LCO)

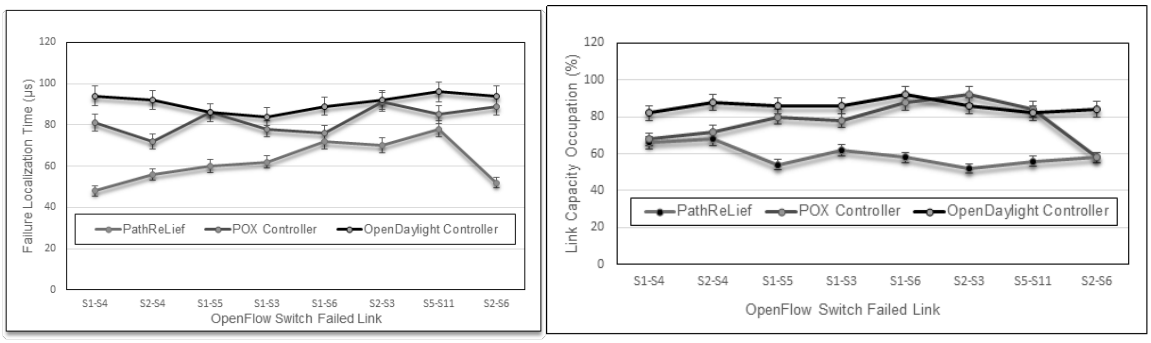
During a link or node failure in the SDN/NFV network in our implementation, some of the available bandwidth is allocated to the backup path. This depends on the network traffic that is to be sent from the failed node/link to the destination node. The LCO is defined as the percentage of the total link capacity allocated to the backup paths considered for forwarding the tagged packets in the SDN/NFV-based network. As shown in Fig. 9b, the PathReLief algorithm provides a minimum link capacity occupation allocated to the backup paths compared to POX and OpenDaylight controllers. This is so because MPTCP and SR enable load balancing between multiple disjointed paths and makes fast transmission of the tagged packets on every link used in the SDN/NFV-based network while guaranteeing the minimum bandwidth [30], [15].

VII. CONCLUSION AND FUTURE WORK

Future softwarized 5G networks have to meet requirements such as resilience, service availability, high system throughput, reliability and ultra-low-latency to support bandwidth-intensive applications and services like live video streaming. To cope with these multimedia services which require an optimized QoE, this paper proposes a QoE-aware MPTCP/SR-enabled architecture, namely, QoEMultiSDN that achieves an optimized E2E QoE-level for the end-users while taking into account the protections and recovery of multiple paths in the network. We employ source routing based on SR and transmit MPTCP video subflows through multiple disjointed shortest paths while ensuring that the end-user's video quality is improved. Our aim is to provide an efficient QoE control and management of video streaming services in future softwarized networks such as 5G. We also propose a multipath protection and link-failure-free MPTCP/SR-based

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access



(a) Failure Localization Time

(b) Link Capacity Occupation

FIGURE 9: The Performance of Failure localization Time and Link capacity occupation

SDN/NFV strategy namely, PathReLief that increases survivability, resilience, availability of services in future softwarized networks. The proposed multipath protection and link failure recovery algorithm reduces greatly the failure recovery time and avoids link congestion in MPTCP/SR softwarized networks. Preliminary results show that, the proposed approach outperforms the MPTCP and regular TCP in terms of system throughput and the end-user's QoE. Our future work will replicate the proposed architecture and consider QoE management and orchestration of resources in future 5G softwarized networks. This work will consider QoE-oriented network sharing/slicing mechanisms by implementing an intelligent scheduling algorithms that can allocate resources among network slices in SDN/NFV. We will also provide extensive experiments and compare QoEMultiSDN and MPTCP SDN-based solutions for DASH proposed in [58] and [59].

REFERENCES

[1] A. Barakabitze, N. Barman, A. Ahmad, S. Zadtootaghaj, L. Sun, M. Martini, and L. Atzori, "QoE management of multimedia services in future networks: A tutorial and survey," IEEE Communications Surveys and Tutorials, vol. 22, pp. 526–565, December 2020.

[2] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017 - 2022,," http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf.

[3] A. Barakabitze, A. Arslan, M. Rashid, and A. Hines, "5G network slicing using sdn and nfv: A survey of taxonomy, architectures and future challenges," Computer Networks, vol. 167, pp. 1–40, February 2020.

[4] I.-H. Mkwawa, B. A. A, and L. Sun, "Video quality management over the software defined networking," IEEE International Symposium on Multimedia (ISM), Dec 2016.

[5] A. A.Barakabitze, L. Madhusanka, and H. Andrew, "QoESoft: QoE Management Architecture for Softwarized 5G Networks," IEEE International Conference on Communications Workshops (ICC Workshops), June 2020.

[6] A. E. Al-Issa, A. Bentaleb, A. B. Alcardo, Z. Thomas, and G. Bogdan, "Bandwidth Prediction Schemes for Defining Bitrate Levels in SDN-enabled Adaptive Streaming ," 15th International Conference on Network and Service Management (CNSM), Oct 2020.

[7] G. Elisavet, A. A. Barakabitze, L. Atzori, L. Sun, and V. Pilloni, "An SDN-approach for QoE management of multimedia services using resource allocation," IEEE International Conference Communications (ICC), pp. 1–7, May 2017.

[8] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using OpenFlow: A survey ," IEEE Communication Surveys Tutorials, vol. 16, no. 1, pp. 493–512, 2014.

[9] A. Doria, J. H. Salim, R. Haas, H. Khosravi, W. Wang, L. Dong, R. Gopal, and J. Halpern, "Internet Eng. Task Force (IETF): Forwarding and Control Element Separation (ForCES)." http://tools.ietf.org/html/draft-smith-opflex-00, March 2010. [Online: Accessed 13-September-2019].

[10] M. Smith, M. Dvorkin, Y. Laribi, V. Pandey, P. Garg, and N. Weidenbache, "OpFlex Control Protocol, Internet Draft, Internet Engineering Task Force," April 2014. [Online: Accessed 11-September-2019].

[11] S. Jain, A. Kumar, S. Mandal, J.Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, and A. Vahdat, "B4: Experience with a globally-deployed software defined WAN,," Proceedings of ACM SIGCOMM,, pp. 3–14, August 2013.

[12] Y. Hong, S. Kandula, R. Mahajan, V. G. M. Zhang, M. Nanduri, and R. Wattenhofer, "Achieving high utilization with software-driven WAN,," Proceedings of ACM SIGCOMM,, pp. 15–20, August 2013.

[13] O. Dobrijevic, M. Santl, and M. Matijasevic, "Ant colony optimization for QoE-centric flow routing in software-defined networks," in Proceedings of the 11th International Conference on Network and Service Management (CNSM), pp. 274–278, November 2015.

[14] A. KasslerA, L. Skorin-Kapov, O. Dobrijevic, M. Matijasevic, and P. Dely, "Towards QoE-driven Multimedia Service Negotiation and Path Optimization with Software Defined Networking ," in 20th International Conference onSoftware, Telecommunications and Computer Networks (SoftCOM), November 2012.

[15] A. A. Barakabitze, "QoE-centric control and management of multimedia services in software defined and virtualized networks," University of Plymouth Research Theses, March 2020.

[16] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "A Novel QoE-Aware SDN-enabled, NFV-based Management Architecture for Future Multimedia Applications on 5G Systems," 8th International Conference on Quality of Multimedia Experience (QoMEX), June 2016.

[17] N. Hyunwoo, D. Calin, and H. Schulzrinne, "Towards Dynamic MPTCP Path Control Using SDN," IEEE NetSoft Conference and Workshops (NetSoft), pp. 286–294, July 2016.

[18] S. Afzal, V. Testoni, C. E. Rothenberg, P. Kolan, and I. Bouazizi, "A Holistic Survey ofWireless Multipath Video Streaming,," arXiv:1906.06184, pp. 1–42, June 2019.

[19] C. Filsfils, N. K. Nainar, C. Pignataro, J. C. Cardona, and P. Francois, "The Segment Routing Architecture ," IEEE Global Communications Conference (GLOBECOM),, February 2015.

[20] J. Wu, C. Yuen, B. Cheng, M. Wang, and J. Chen, "Streaming High-Quality Mobile Video with Multipath TCP in Heterogeneous Wireless Networks," IEEE Transactions on Mobile Computing, vol. 15, pp. 2345–2361, March 2016.

[21] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "A Novel QoE-Centric SDN-based Multipath Routing Approach of Mutimedia Services over 5G Networks," IEEE International Conference on Communications,, May 2018.

[22] "MPTCP Linux Kernel Implementation." [Availableat]:http://www.multipath-tcp.org/.

[23] N. Kukreja, G. Maier, R. Alvizu, and A. Pattavina, "SDN based Automated Testbed for Evaluating Multipath TCP," IEEE International Conference on Communications Workshops (ICC),, pp. 718–723, May 2016.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

[24] L. Luo, H. Yu, and S. Luo, "Achieving Fast and Lightweight SDN Updates with Segment Routing,," IEEE Global Communications Conference (GLOBECOM),, pp. 1–6, December 2016.

[25] B. Stephens, A. Cox, W. Felter, C. Dixon, and J. Carter, "Past: Scalable ethernet for data centers," Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies, CoNEXT, pp. 49–60, December 2012.

[26] X. Corbillon, R. Aparicio-Pardo, N. Kuhn, G. Texier, and G. Simon, "Cross-layer scheduler for video streaming over MPTCP,," Proceedings of the 7th International Conference on Multimedia Systems,, May 2016.

[27] C. James, E. Halepovic, M. Wang, R. Jana, and N. K. Shankaranarayanan, "Is Multipath TCP (MPTCP) Beneficial for Video Streaming over DASH?," IEEE 24th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems,, pp. 331–336, September 2016.

[28] L. Davoli, L. Veltri, P. L. Ventre, G. Siracusano, and S. Salsano, "Traffic Engineering with Segment Routing:SDN-based Architectural Design and Open Source Implementation," Fourth European Workshop on Software Defined Networks (EWSDN),, September 2015.

[29] A. A. Barakabitze, I.-H. Mkwawa, L. Sun, and E. Ifeachor, "QualitySDN: Improving Video Quality using MPTCP and Segment Routing in SDN/NFV," IEEE Conference on Network Softwarization, June 2018.

[30] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "Multipath Protections and Dynamic Link Recovery in Softwarized 5G Networks using Segment Routing," EEE GLOBECOM 2019 Workshop on Future Internet Architecture, Technologies and Services for 2030 and beyond, Dec 2019.

[31] S. Balazs, N. Felician, C. Levente, L. Gulyás, and A. Gulyás, "SDN based Testbeds for Evaluating and Promoting Multipath TCP," In Proceedings of IEEE ICC- Next-Generation Networking Symposium,, pp. 3044–3050, June 2014.

[32] J. Duan and C. W. Zhi Wangy, "Responsive multipath TCP in SDN-based datacenters," IEEE International Communication Conference (ICC), June 2015.

[33] K. D. Joshi and K. Kataoka, "SFO: SubFlow Optimizer for MPTCP in SDN," 26th IEEE International Telecommunication Networks and Applications Conference (ITNAC) ,, pp. 173–178, December 2016.

[34] S. Chung, S. Moon, and S. Kim, "The virtualized MPTCP proxy performance in cellular network," Ninth International Conference on Ubiquitous and Future Networks (ICUFN), pp. 703–707, July 2017.

[35] T.-T.-L. Nguyen, T.-M. Pham, and H. T. T. Binh, "Adaptive multipath routing for network functions virtualization," Proceedings of the 7th Symposium on Information and Communication Technology, pp. 222–228, December 2016.

[36] Y. Liu, Y. Li, M. Caniniy, Y. Wang, and J. Yuan, "Scheduling multi-flow network updates in Software-Defined NFV systems," IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), April 2016.

[37] M. Sandri, A. Silva, L. A. Rocha, and F. L. Verdi, "On the Benefits of Using Multipath TCP and Openflow in Shared Bottlenecks," IEEE 29th International Conference on Advanced Information Networking and Applications, April 2015.

[38] S. Tariq and M. Bassiouni, "Packet-In Message Control for Reducing CPU Load and Control Traffic in OpenFlow Switches," In Proceedings of IEEE CCNC,, January 2015.

[39] S. Zannettou, S. Michael, and P. Fragkiskos, "Exploiting path diversity in datacenters using MPTCP-aware SDN," IEEE Symposium on Computers and Communication (ISCC),, pp. 539–546, June 2016.

[40] C. Thaenchaikun and W. Panichpattanakul, "Control-plane OpenFlow segment routing SDN for network control messages reduction,," 14th IEEE International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) ,, pp. 322–325, June 2017.

[41] G. Trimponias, Y. Xiao, H. Xu, X. Wu, and Y. Geng, "On Traffic Engineering with Segment Routing in SDN based WANs,," IEEE/ACM Trans. Networking,, vol. 25, pp. 1–17, September 2017.

[42] M.-C. Lee and J.-P. Sheu, "An efficient routing algorithm based on segment routing in software-defined networking," Journal of Computer Networks,, vol. 103, pp. 44–55, July 2016.

[43] I. B. Akbari, D. Krisnandi, R. Wardoyo, and D. Syamsi, "Sub-second path restoration for stateful-based segment routing," International Conference on Computer, Control, Informatics and its Applications (IC3INA), Nov 2018.

[44] A. Sgambelluri, S. S. S. Anna, A. Giorgetti, F. Cugini, F. Paolucci, and P. Castoldi, "Openflow-based segment protection in ethernet networks," IEEE/OSA Journal of Optical Communications and Networking, vol. 5, no. 9, pp. 1066–1075, 2013.

[45] Z. Cheng, Z. Xiaoning, Y. Li, S. Yu, R. Lin, and L. He, "Congestion-aware local reroute for fast failure recovery in software-defined networks," IEEE/OSA Journal of Optical Communications and Networking, vol. 9, no. 11, pp. 934–944, 2017.

[46] v. A. Niels L.M, B. J. van Asten, and F. A. Kuipers, "Fast recovery in software-defined networks," Third European Workshop on Software Defined Networks, Sept 2014.

[47] Y. Xiong, Y. Li, B. Zhou, R. Wang, and G. N. Rouskas, "Sdn enabled restoration with triggered precomputation in elastic optical interdatacenter networks," IEEE/OSA Journal of Optical Communications and Networking, vol. 10, no. 1, pp. 24–34, 2018.

[48] T. H, R. Schatz, E. Biersack, and L. Plissonneau, "Internet video delivery in YouTube: from traffic measurements to quality of experience," Data Traffic Monitoring and Analysis. Springer, pp. 264–301, June 2013.

[49] A. K. Saha, K. Sambyo, and C. Bhunia, "Topology Discovery, Loop Finding and Alternative Path Solution in POX Controller," Proceedings of the International MultiConference of Engineers and Computer Scientists, 2016.

[50] "ETSI GS NFV-MAN 001 V1.1.1: Network Functions Virtualisation (NFV); Management and orchestration," ETSI Ind. Spec. Group (ISG) Netw. Functions Virtualisation (NFV),," pp. 1–183, December 2014.

[51] F. Ongaro, E. Cerqueira, L. Foschini, A. Corradi, and M. Gerla, "Enhancing the Quality Level Support for Real-time Multimedia Applications in Software-Defined Networks ,," International Conference on Computing, Networking and Communications, Communication QoS and System Modeling Symposium,, pp. 505–509, June 2015.

[52] H. Shen, Y. Liu, T. Wang, H. Yang, and L. Sang, "QoE-optimal rate adaptation for HTTP adaptive streaming," IEEE/CIC International Conference on Communications in China (ICCC), July 2016.

[53] Y. Liu and S. Dey, "User experience modeling for dash video," IEEE Packet Video Workshop, pp. 1–8, July 2013.

[54] ITEC, "Dynamic Adaptive Streaming over HTTP:Dash plugin for vlc,," http://www-itec.uni-klu.ac.at/dash/?page_id=10, June 2012.

[55] E. Thomas, M. O. van Deventer, T. Stockhammer, A. C. Begen, and J. Famaey, "Enhancing MPEG DASH Performance via Server and Network Assistance," SMPTE Motion Imaging Journal, vol. 126, pp. 22–27, Jan 2017.

[56] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over http," In ACM SIGCOMM, pp. 325–338, August 2015.

[57] J. P. Vasseur, M. Pickavet, and P. Demeester, "Network Recovery: Protection and Restoration of Optical, SONETSDH, IP and MPLS: San Francisco, California: Morgan Kaufmann," 2004.

[58] H. Korhan, S. K. Reza, C. Cihat, and S. Muge, "Towards QoS-aware Routing for DASH Utilizing MPTCP over SDN," IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), pp. 1–6, December 2017.

[59] L. Guillen, S. Izumi, T. Abe, and T. Suganuma, "Sand/3: Sdn-assisted novel qoe control method for dynamic adaptive streaming over http/3," Electronics, vol. 8, Dec 2019.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3039953, IEEE Access

ALCARDO ALEX BARAKABITZE received a Ph.D. degree in Computing and Communications from the University of Plymouth, UK. He received the degree in Computer Science with Honours from the University of Dar es Salaam, Tanzania in 2010 and Master Degree of Electronics and Communication Engineering with first class from Chongqing University, PR China, in May 2015. He has worked as Marie Curie Fellow (2015–2018) in MSCA ITN QoE-Net. Dr. Barakabitze has also worked as an IRC fellow at UCD. Dr. Barakabitze is recognised as the 2015 outstanding International Graduate Student of Chongqing University, China due to his excellent performance. He was a visiting researcher in the Department of Electrical and Electronics Engineering, University of Cagliari, Italy and the ITU-T-Standardization Department in 2016 and 2017 respectively. He has numerous publication in International peer-reviewed conferences and journals. Dr. Barakabitze has served as session chair of Future Internet and NGN Architectures during the IEEE Communication Conference in Kansas City, USA. He was also the Keynote and Panel Chairs at the International Young Researcher Summit on Quality of Experience in Emerging Multimedia Services (QEEMS 2017), that was held from May 29–30, 2017 in Erfurt, Germany. Dr. Barakabitze is a Reviewer for various journals and serves on technical program committees of leading conferences focusing on his research areas. His research interests are 5G, SDN and NFV, Quality of Experience (QoE), network management, video streaming services and ICT4D.

IS-HAKA MKWAWA received a PhD degree in Computing from the University of Bradford, UK in 2004. He is now a lecturer of Computer Systems and Networks with the School of Engineering, Computing and Mathematics at the University of Plymouth, UK. He is also an honorary lecturer of Computing with the Department of Computer Science at the University of Liverpool and a visiting professor at the College of Computer Science and Information Engineering, Sudan University of Science and Technology. His main research interests include Wireless Networks, Multimedia Communication, QoE Management and Control, Next-Generation Internet, Analytical Modelling, Cloud Computing, and Big Data. Dr Mkwawa holds a research track record of publications in the top international journals and conferences. His publications including "Guide to Voice and Video over IP: For Fixed and Mobile Networks (Springer, 2013)" book, 14 book chapters. He has served as the committee member of various professional conferences/workshops and independent external member of degree approvals for BSc and MSc Computer Networking programmes. Dr Mkwawa is a member of the IEE Computer and Communication societies and he is a fellow of the Higher Education Academy.

ANDREW HINES is an Assistant Professor with the School of Computer Science, University College Dublin, Ireland. He leads the QxLab research group and is an investigator in the Science Foundation Ireland CONNECT and INSIGHT research centres, that are leaders in future networks and machine learning and data analytics respectively. He was awarded the US-Ireland Research Innovation award in 2018 by the Royal Irish Academy and American Chamber of Commerce for leading collaboration with Google since 2012 that has led to two patents, 10 industry collaborative research papers and three technology licences. The technology developed is currently used at Google for QoE modelling and testing QoE for the 30,000 most popular videos per day in YouTube. Dr Hines is a senior member of the IEEE and a leading expert in Quality of Experience for media technology. Dr Hines represented Ireland on the management committee of the European COST action on Quality of Experience, Qualinet. His primary research interests are in video signal processing, 5G, SDN and network management and machine learning for data driven quality of experience prediction across a variety of domains.

LINGFEN SUN (M'02) received a Ph.D. degree in Computing and Communications from the University of Plymouth, UK, where she is an Associate Professor (Reader) of multimedia communications and networks with the School of Engineering, Computing and Mathematics. She has led the Plymouth team in several EU FP7/H2020 and industry funded projects related to multimedia communications and QoE, including ADAMANTIUM, GERYON and QoENet. She has authored or coauthored over 100 peer refereed technical papers since 2000 and filed one patent. Her current research interests include multimedia quality assessment, QoS/QoE management , cloud computing, future Internet and AI/ML applications. She was an Associate Editor of the IEEE Transactions on Multimedia (2016 – 2018), Symposium Chair for Communication Software, Services and Multimedia Applications of IEEE ICC 2019/2014, Vice-Chair of IEEE MMTC (2018 – 2020), and the Founding Chair of QoE Interest Group of IEEE MMTC (2010- 2012). She is a Fellow of the Higher Education Academy.

EMMANUEL IFEACHOR is a professor of intelligent electronics systems and Head of Signal Processing and Multimedia Communications Lab at the University of Plymouth, UK. He received the B.Sc. (Hons) degree in communication engineering from the University of Plymouth in 1980, the M.Sc. degree and DIC in communication engineering from Imperial College, London in 1981, and a PhD degree in medical electronics from the University of Plymouth in 1985. His main research interests are in signal processing, data science, artificial intelligence and machine learning techniques and their applications to important real-world problems in biomedicine, multimedia communications and audio. He has published extensively in these areas, including co-authoring a major text book, Digital Signal Processing - A Practical Approach (Prentice Hall, 2002). Professor Ifeachor has successfully led many industry and government funded projects. He previously served as Head of School at Plymouth (School of Electronic, Communication and Electrical Engineering).

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/