IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 29 June 2023, accepted 17 August 2023, date of publication 29 August 2023, date of current version 5 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3309599

RESEARCH ARTICLE

QoE-Aware Dynamic Resource Management in Future Softwarized and Virtualized Networks

ALCARDO ALEX BARAKABITZE[1], (Member, IEEE), IS-HAKA MKWAWA[2], (Member, IEEE), ANDREW HINES[1], (Senior Member, IEEE), AND RAY WALSHE[3], (Senior Member, IEEE)

[1]School of Computer Science, University College Dublin, Dublin 4, D04 V1W8 Ireland
[2]School of Engineering, Computing and Mathematics, University of Plymouth, PL4 8AA Plymouth, U.K.
[3]School of Computing, Dublin City University, Dublin 9, D09 V209 Ireland

Corresponding author: Alcardo Alex Barakabitze (alcardo.barakabitze1@ucd.ie)

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme through the Marie Skłodowska-Curie Grant under Agreement 801522; and in part by the Science Foundation Ireland under Grant 12/RC/2289_P2, Grant 13/RC/2077_P2, and Grant 13/RC/2106_P2.

ABSTRACT With the advent of multimedia streaming services such as 4K/8K and extended reality, managing and optimizing multimedia service delivery to end-users remains a challenge for mobile network operators, Internet service providers, and over-the-top (OTT) providers. To solve the present concerns and challenges associated with multimedia traffic management and quality of experience (QoE) in next-generation networks, new network and service management approaches for multimedia services are required. Network softwarization and virtualization paradigms that leverage software defined networking (SDN) and network function virtualization (NFV) are considered as key technologies for network and service management in future softwarized networks. This paper adopts SDN control logic to implement a dynamic QoE-aware resource management approach in future softwarized and virtualized networks. We propose QoESoft, a QoE-aware network softwarization approach for multimedia streaming services that (a) performs a dynamic link and switch resources management, and (b) improves end-user' QoE in softwarized systems. We introduce two important components, the QoE-sdnFlow monitor and QoE-sdnFlow manager as an extension to the SDN controller to monitor and manage the overall utilization of resources of mapped virtual links and nodes. We present a practical implementation of the QoESoft approach using a dash.js reference player where the bitrate decision adaptation logic and video segment download process are modified by introducing two components: the *BandwidthPredictor* and the *Reporting* functions. The *BandwidthPredictor* component ensures that the video bitrate is selected based on the available resources on the client's side in a dynamic manner. The *Reporting* module provides the current streaming status of the dash.js player (e.g., stall duration, bitrate switching, startup delay, and video QoE). In addition, we present the initial evaluations of virtual network survivability and an economical analysis that maximizes the profit for OTTs/ISPs by providing better QoE to customers. The performance of the proposed approach is analyzed through extensive experiments using a Laboratory for Image and Video Engineering - Amazon Prime Video (LIVE-APV) Streaming Database running over the developed softwarized - DASH- based platform based on the Mininet and POX controller. Preliminary results indicate that *QoESoft* outperforms the baseline approach in terms of link resources utilization and switch resources utilization, low-live latency, startup delays, bitrate switching, stall duration and video QoE.

INDEX TERMS SDN, QoE, 5G/6G, multimedia streaming services, NFV, network softwarization, video quality, network management.

I. INTRODUCTION

The demand for streaming video content at higher resolutions with good quality of experience (QoE) is increasing at

The associate editor coordinating the review of this manuscript and approving it for publication was Yuan Gao.

a faster pace towards the development of next -generation wireless networks (e.g., sixth-generation mobile/wireless networks (6G)) [1], [2]. HTTP adaptive video streaming (HAS) is now the common streaming technology for providing the end-users QoE [3], [4]. However, service providers are very challenged in offering high QoE to their users in future mobile/wireless networks, as explored by Barakabitze et al. [5]. Network softwarization using network function virtualization (NFV) and software-defined networking (SDN) will provide the flexible programmatic operation of the control plane required by mobile network operators and service providers to adapt to changing network conditions while streaming video content (e.g., 4K/8K/12K, 360°) with QoE guarantees [6], [7], [8], [9].

SDN is an approach to network management that makes a network programmable, dynamic, and adaptable to changing network conditions, and provides an efficient network configuration to improve network performance and monitoring [5], [6]. NFV is a network architecture that virtualizes all classes of network functions and provides the timely delivery of end-to-end services at a lower cost (e.g., reduces capital expenditure (CAPEX) and operating expense (OPEX)). NFV also enables scaling of the network infrastructure capacity to support the huge demand for emerging multimedia streaming services in future networks. One or multiple service providers in softwarized networks can use substrate networks (SNs) to offer multimedia streaming services to customers and provide high QoE through virtual networks (VNs). Currently, SDN and NFV are steering the "Softwarization" and virtualization of future telecommunication [6] systems including 6G and beyond networks by enabling the rapid deployment of new business applications, mobility management, and traffic rerouting for congestion avoidance in the network [2]. SDN and NFV radically the design of 5G and beyond networks by offering features such as reconfigurability, programmability, flexibility and adaptability to support variety of use cases including multimedia streaming services.

Network virtualization (NV) refers to abstracting network resources that are traditionally delivered from hardware to software [10]. 5G network virtualization provides a mechanism that allows multiple VNs to share resources with one or more SNs. These resources for any given VN are completely isolated from the others, and appear to belong to different physical networks. The shared resources can be leased by the VN operators to other VNs. They can also use resources to provide services to end-users and allow them to establish multiple flexible networks that are driven by the end-user's QoE requirements in the network. SNs can be managed and controlled by infrastructure providers (InPs) [11]. However, because InPs can host multiple VNs and support their complete isolation, the only challenge is to allocate and manage resources efficiently to multiple VNs. Traffic and network control are InP requirements for future softwarized networks.

Future networks [12], such as 5G/6G need to support different services (e.g., 360° videos, video gaming, 4K/8K/12K) with high quality. Dynamic network resource (e.g., link, switch memory, bandwidth resources) management is a major problem in multimedia communication networks [13]. As the amount of multimedia traffic in future softwarized networks increases, forwarding flow rules along each substrate delivery path in SDN switches will also increase. A larger number of flow rules would in turn increase the load at the controller and SDN switches that employ ternary content-addressable memory (TCAM) to store forwarding rules [14]. TCAM is power hungry and practically supports a limited number of 2k - 20k forwarding rules [15]. Therefore it is very important to consider the management of network resources (e.g., switch and link resources) in softwarized networks. This is very crucial because the switch memory in SDN is power hungry and expensive [15].

The challenge with dynamic resource management [16] in virtualized networks is that different services (multimedia streaming services) not only require different QoE guarantees but also different resource types (e.g., bandwidth and queue size) are segmented into many links and nodes in the networks. Most of the works explored in the literature regarding dynamic resource management employs three mechanisms, namely: (a) performance dynamics modeling [17], (b) control theory [18], [19], and (c) workload prediction [20]. Some of these works, for example, [18] and [19] guarantee QoS and perform QoS management in self-managed and controlled systems. This is achieved only in web application services and does not consider the QoE aspects in the context of future softwarized networks. In addition, these studies do not consider QoE-aware resource management of links and switches and the cost of VN resources in softwarized networks [21].

In this article, we extend our previous concept/works [7], [22] to QoE-aware SDN-enabled, NFV-based management architecture for managing future multimedia streaming services in softwarized systems. We build on these works and propose *QoESoft*, a QoE-aware network softwarization strategy in future softwarized networks that controls and manages emerging multimedia streaming services through a dynamic switch and link resources management. While previous works [23], [24] considered only link resources such as bandwidth, *QoESoft* considers both VN node resources (memory) and resource costs. *QoESoft* is implemented as an extension of the POX controller where two important modules are introduced, namely: the *QoE-sdnFlow monitor* and *QoE-sdnFlow manager*. These two entities are responsible for monitoring the average load of the substrate switches and link and resource costs of the mapped virtual links. They also control and perform dynamic management of the end-user's QoE during video streaming. The *QoE-sdnFlow manager* uses the information of the arrivals and departures of VN requests to modify (add and/or delete) flow rules in SDN switches. It is worth mentioning that the arrival and departure of VN requests are performed dynamically, leading to better resource utilization efficiency in future softwarized networks. Dynamic QoE-aware resource management also enables the delivery of multimedia streaming services in softwarized networks with a good QoE [5], [7].

The proposed *QoESoft* will enable service providers (SPs), mobile network operators (MNOs), and over-the-top (OTTs) to deliver high quality multimedia streaming services to their customers using QoE-aware resource management and a software life-cycle based management approach in softwarized systems. The *QoESoft* approach overcomes the key QoE provisioning limitations in traditional networks such as: (a) the inability of current networks to adapt to changing end-user's demands, and/or network transmission/traffic (because of the increasing multimedia streaming services), (b) the complexity of network management in SDN/NFV systems, (c) lack of network control and sub-optimal utilization of network resources, and (d) provision of many QoS tools and manual configuration of network services. To the best of our knowledge, this is among the first studies to examine dynamic QoE-aware resource management mechanisms in future softwarized networks. The contributions of this paper are five-fold:

- We present QoESoft, a dynamic QoE-aware resource management mechanism for managing multimedia streaming services such as standard definition (SD) and high definition (HD) videos in future softwarized networks.
- We implemented two components, the QoE-sdnFlow monitor and QoE-sdnFlow manager as an extension to the SDN controller to monitor and manage the average substrate switches and links. These entities are used to manage the resource costs of mapped virtual links in softwarized networks.
- We present a practical implementation of QoESoft using the dash.js [25] reference player by introducing two important functions: the *BandwidthPredictor* and *Reporting* in the dash.js player. The *Reporting* function provides the current streaming status of the dash.js player such as the live latency, stall duration, selected bitrate, bitrate switching, startup delay, and video QoE. The *BandwidthPredictor* module dynamically ensures that the bitrate is selected based on the available resources on the client's side. We also changed the bitrate decision adaptation logic and video segment download process by ensuring that the video bitrate levels are selected at the beginning of every segment downloaded.
- We provide the initial evaluations of a virtual network survivability and QoE-based pricing models that maximize the profit for OTTs/ISPs by providing better QoE to customers.
- We present preliminary results to show the credibility and innovativeness of the QoESof approach through extensive DASH experiments using standard definition (SD) and High definition (HD) video streaming from the Laboratory for Image and Video Engineering - Amazon Prime Video (LIVE-APV) streaming database.

The rest of this paper is structured as follows: Section II presents related work. Section III provides the resource management problem in softwarized and virtualized networks and Section IV presents the dynamic QoE-aware resource management system in softwarized networks. Section V provides the network model and QoE-aware resource management algorithm for future softwarized networks. The performance and evaluation of the proposed strategy are presented Section VI while the conclusion is presented in Section VII.

II. RELATED WORK

Dynamic QoE resource management and resource allocation for future multimedia streaming services in softwarized networks have been explored in recent years [5], [26], [27], [28], [29], [30], [31], [32], [33], [34]. Barakabitze et al. [7] introduced a QoE-softwarized and virtualized architecture that leverage SDN and NFV for managing future multimedia streaming services in 5G networks.

Farahani et al. [27] proposed ARARAT, a multi-layer architecture that leverages SDN, NFV, and edge computing paradigms to minimize HAS clients' serving time and network cost while considering available resources and all possible serving actions. A total of 250 HAS players made up the large-scale cloud testbed used by the authors for their research. In comparison to state-of-the-art methods, the experimental results show that ARARAT improves user's QoE by at least 47%, reduces streaming costs–including bandwidth and computation costs by at least 47%, and increases network utilization by at least 48%. However, this study does not consider the link and switch costs.

Benzad and Taleb [35] proposed an E2E network and service management architecture that enables the management and automation of emerging and future networks and services in a flexible, efficient, and qualitative manner. Ahmad et al. [34] proposed Timber, an experimental tool for the design and evaluation of QoE management and monitoring procedures for video streaming. The main functionalities of the Timber include: an SDN application that considers QoE-aware management decisions and a user-end probe at the client video player that monitors QoE-related video parameters during video streaming.

Majdabadi et al. [28] proposes SODA-Stream, an SDN-based optimization framework for enhancing the QoE of end-users in DASH -driven streaming by maximizing the number of concurrent streaming sessions that can be accommodated in a network. The practical SDN -based implementation of the framework employs dynamic routing and bandwidth allocation for video streaming sessions. The findings show that SODA-Stream significantly outperforms traditional network routing and resource allocation algorithms, (a) accepting 52% more sessions,(b) improving bandwidth allocation by 45%, (c) reducing bandwidth waste by 70%, and enhancing playback and watching quality.

Bentaleb et al. [33] proposed SDNHAS, an enhanced intelligent streaming architecture that leverages SDN capabilities to maximize QoE by assisting HAS players in making better streaming adaptation decisions. SDNHAS provides intelligent adaptive streaming delivery and network management with the highest level of viewer satisfaction among heterogeneous HAS players. This is achieved using a central entity that has a broad overview of all competing players and can monitor player interactions and gather statistics [33]. Based on the SSIMplus-based clustering method, the SDN-based external application chooses the ideal bitrate and video quality that can equitably optimize viewer QoE while correctly allocating the available bandwidth between active streaming sessions. However, SDNHAS does not perform well in a scenario of multi-bottleneck shared networks and different patterns of cross traffic dynamics [33]. The authors in [32] proposed SQAPE, a novel QoE-aware path selection algorithm for large-scale SDN-based mobile networks that predicts video streaming performance and QoE using QoS measurements in the network. The proposed path selection algorithm that runs on the SDN architecture is based on the Dijkstra shortest-path algorithm adaption. The distance metric in SQAPE is a combination of the MOS estimate and link utilization contribution [32]. However, SQAPE does not consider video streaming flow data or client/server-side information.

Barakabitze et al. [36] proposed QoEMultiSDN, a novel QoE-aware management mechanism that uses multiPath-TCP/segment-routing (MPTCP/SR) techniques to explore information from both client and network sides in SDN/NFV networks. To optimize E2E DASH service delivery to users, QoEMultiSDN employs QoE-based multisource routing and QoE optimization, and (b) multipath protection and dynamic link-failure-free approaches. Sun et al. [37] proposed intelligent strategies for homogeneous networks based on SDN to handle control and subscriber management in 5G systems. The authors developed a variety of SDN strategies for load balancing, traffic prediction, user density prediction and radio resource allocation in 5G networks. Barakabitze et al. [22] adopted an SDN control platform to implement a QoE-aware resource management mechanism in future softwarized 5G networks.

The network softwarization approach proposed in [22] for multimedia streaming services that performs dynamic resource management in softwarized 5G systems achieves a better performance by 78% in terms of resource utilization compared to the baseline approach. The authors in [38] proposed a machine learning (ML) strategy using multi-agents to perform virtual network resource management in SDN/NFV-based networks. The proposed approach employs agents that allocate network resources to virtual nodes and links using evaluation feedback to learn the optimal policy of the required resources. Mijumbi et al. [39], [40] introduced a graph neural network-based approach that can predict future resource requirements for each VNF component (VNFC) by exploiting the VNF forwarding graph topology information. The virtualized IP multimedia subsystem, and real VoIP traffic traces were used to evaluate the performance of the proposed approach. The results indicate that the proposed strategy reduces the average number of dropped calls by at least 27% and improves call setup latency by over 29% while the average prediction accuracy is 90% compared to 85% when resources are allocated manually and/or statistically. However, these approaches have not been implemented or tested for multimedia streaming services.

Alfoudi et al. [48] proposed a network slicing resource management (NSRM) approach that assigns resources required for each network slice in a long term evolution (LTE) network while considering the isolation of resources among different slices. The proposed mechanism ensures fair resource sharing of the distributed bandwidths between users belonging to the same slice. Erfanian et al. [45] employed two types of VNFs, (a) a virtual transcoder function (VTF), and (b) a virtual reverse proxy (VRP) to propose an Optimizing reSourCe utilizAtion in live video stReaming (OSCAR). OSCAR provides advanced video coding (AVC)-based live streaming services in networks. The authors in [23] proposed reactive VN resource management techniques for dynamic resource management in softwarized networks. A single or multiple nodes and their associated VN links are migrated using the proposed reconfiguration algorithm to control migration costs. However, this is done under the condition of failed link mapping and does not consider switch loading.

Liotou et al. [31] proposed a programmable QoE-SDN APP that enables MNOs to dynamically provide QoE -aware network capability exposure feedback to the corresponding virtual service providers based on video bitrate prediction and mobility mechanisms. QoE assessment logic is the main component of the QoE-SDN APP that determines the QoE per application based on specific KPIs (e.g., stalling events during video streaming). It also provides guidance for VSP regarding video streaming caching strategies and encoding bitrates. However, the QoE-SDN APP does not consider the link and switch resource utilization and virtual network survivabilitys as well as the cost of VN resources in softwarized networks.

Martin et al. [49] provided an autonomous QoE -aware network resource allocation approach that can predict the topology setup and amount of network resources that can be allocated to users based on traffic demands over 5G systems. Live and on-demand dynamic adaptive video streaming over HTTP and high efficiency video coding services are employed for evaluation. However, this approach does not evaluate the economic analysis of the transmitted video flows in terms of costs, income and profits. It also does not consider virtual network survivability mechanisms or the overall utilization of link/switch resources.

Zhu et al. [50] proposed a QoE-aware 3D video streaming in SDN enabled mobile edge computing to allocate resources and reduce delays based on the information collected during 3D video playback. To optimize the various QoE goals for users while adapting t the dynamic changing bandwidth conditions and viewports, the MEC servers provide computing and caching resources. Thus, the time to respond to the end-user's requests is significantly reduced. This approach does not consider live video streaming in the context of VN. It does

TABLE 1. A comparison of QoE-aware dynamic resource management approaches and QoESoft.

Approach	Real-Time Video Streaming	QoE-Consideration	Objectives/Functionality	Softwarized Network
Barakabitze *et al.* [22],Grigoriou *et al.* [41], [30]	Yes	Yes	QoE management of multimedia streaming services [42] using resource allocation and bandwidth prediction schemes in SDN adaptive video streaming networks [43]. To provide an optimized E2E QoE for multimedia service delivery using MPTCP and SR techniques by acquiring information from both the client and network side.	NO
[7], [37], [35], [44], [45]	No	Yes	To improve the QoE of multimedia streaming services through the fulfilment of personalized QoE application requirements in softwarized networks [43]. These papers present also intelligent SDN-based models that can efficiently manage heterogeneous network infrastructure and resources [37]. In addition, they also provide QoE-awareness and optimization mechanisms for network resources utilization in live video streaming in softwarized networks [44], [45]. However, no practical implementation is given in these works.	Yes
[46], [38], [40], [39], [23], [24], [47], [48]	No	No	To provide an efficient resource management for network slicing and an integrated allocation of link bandwidth and flow table for multiple control applications in SDN. Some of these approaches perform dynamic resource management and prediction of VNFs resource requirements in softwarized and virtualized networks [39], [40].	Yes
[29], [27], [26], [28]	No	Yes	Collaborative QoE management using SDN and multi-access edge assisted architectures for enhancing the end-user's QoE [29], [26], [27]. Some works introduce SDN-based optimization framework for maximizing the number of concurrent video streaming sessions and maximize streaming video quality [28].	Yes
QoESoft	Yes	Yes	To present a dynamic QoE-aware resource management mechanism for multimedia streaming services (SD and HD) in future softwarized networks. The *QoE-sdnFlow monitor* and *QoE-sdnFlow manager* are implemented as an extension to the POX controller to monitor and manage the average substrate switches and links. Practical implementations of the dash.js [25] reference player is given by changing the bitrate decision adaptation logic and video segment download process. QoESoft ensures that video bitrate levels are selected at the beginning of every segment downloaded using the $BandwidthPredictor$ and $Reporting$ modules.	Yes

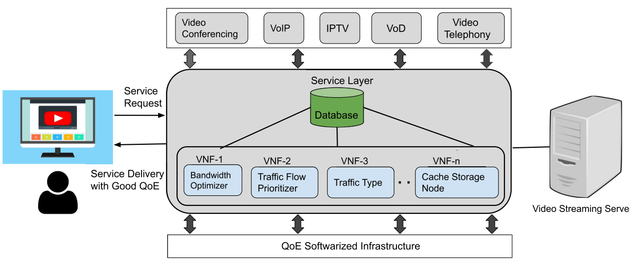


FIGURE 1. End-to-end multimedia streaming use case.

not also take into account link/switch resource utilization and virtual network survivability in future softwarized networks.

To this end, we propose a dynamic QoE-aware resource management mechanism for managing multimedia streaming services (SD and HD videos) in softwarized networks. QoE-Soft improves the QoE to customers while considering the link and switch resources management and virtual network survivability. The goal is to enable SP, MNOs and OTTs to offer better QoE to end users through intelligent and flexible techniques for regulating and managing all available network resources in softwarized networks. Table 1 provides a comparison of the QoESoft architecture with other approaches in the literature.

III. PROBLEM FORMULATION AND DESCRIPTION

A. E2E NETWORK VIDEO STREAMING USE CASE

Fig. 1 shows a video streaming scenario for softwarized networks. We make an assumption that a chain of virtual network functions provide services based on user's demands with better quality. Multimedia streaming services such as video on demand (VoD), IPTV, video conferencing etc. are offered at the application layer. The video requests from the end-users are transmitted across these virtual network functions. The VNFs forms a service delivery graph for establishing the VNFs forwarding graph (VNFFG). The VNFFG defines the chain of VNFs which are linked together to instantiate a network service. The VNFFG basically shows the graph of logical links connecting VNF nodes to describe the flow of network traffic between VNFs. The database stores the VNF-FG with their associated information of resources on all virtual links/switches for the end-users. The only challenge during multimedia streaming is how to manage both switch and link resources for VNs that are mapped to SN while considering the user's QoE.

B. VIRTUAL NETWORK ASSIGNMENT PROBLEM DESCRIPTION

Fig. 2 illustrates the resource allocation problem in virtual networks which consists of two main steps: virtual network embedding (VNE) where VNs are embedded in SN and dynamic network resource management. The virtual network provider specifies the resource requirements for both the nodes and links to the SN provider. We use the network indicated in Fig. 3 to describe the VN to SN mapping problem in future softwarized networks. Consider an *ABC* that is created as a virtual network to be mapped onto SN $\{S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S5 \rightarrow S6\}$. In this case, each virtual switch $i \in (A, B, C)$ should be mapped to the substrate SDN switch $m \in \{S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S5 \rightarrow S6\}$ with connected links and enough switch memory of sufficient bandwidth. This is done to support the virtual links in softwarized

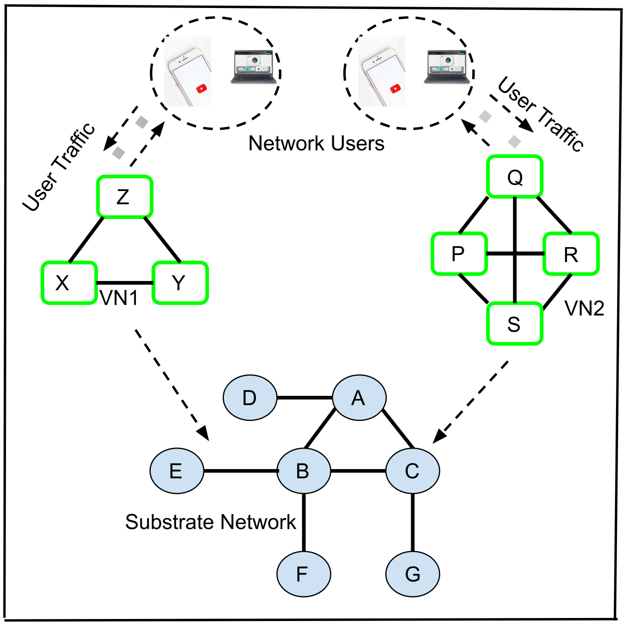


FIGURE 2. Virtual network resource allocation.

networks that are responsible for delivering the multimedia streaming services to users. The link and node assignment mechanisms described in section V-A enable the mapping of VN to SN.

However, in future softwarized networks, such mapping of virtual nodes and links to the corresponding substrate nodes and links may cause some issues. For example, consider the assignment of virtual links AC and AB to substrate paths $\{S1 \rightarrow S6 \rightarrow S4\}$ and $\{S1 \rightarrow S6 \rightarrow S2\}$ respectively. Substrate switch $S6$ acts as the bridging point for multimedia streaming flows in softwarized networks. To achieve this assignment, the virtual network layer needs to define forwarding flow rules in virtual switch A to direct network traffic to SDN switches B and C. It is worth noting that extra rules are required at the network layer for each flow in the switch. This ensures that the defined forwarding flow rules in the switches arrive to their intended destination in softwarized networks.

Although path splitting approaches have been shown to provide better resource utilization in VNs when used for virtual link mapping, this would create issues in softwarized networks. This is because each sub-flow requires forwarding flow rules along the substrate path in every SDN switch when a virtual flow is split into multiple sub-flows. As described above, this would in turn require more TCAM, which (a) consumes a lot of power, (b) is expensive to build, and (c) dissipates a high level of heat [46]. Moreover, path splitting would also result in high resource fragmentation at the substrate network layer when performing virtual network- to -substrate resource mapping in a static manner. The vision of future softwarized networks [1] is to enable the delivery of multimedia streaming services (4K/8K/12K, 360°), personalized TV etc.) through logical network slices to meet specific user's demands using artificial intelligence (AI) and big data analytics for shaping the video QoE [1].

Because of the large number of forwarding flow rules for a single flow in a softwarized network, VN owners incur high costs for their network slices. A large number of rules may also cause VN requests to be rejected because of the SN switch memory depletion. Therefore, it is essential to balance the number of rules in future softwarized networks to avoid overutilization or underutilization of available resources and reduce network congestion during multimedia streaming. This would lead to an optimized QoE of transmitted multimedia streaming services to end-users in softwarized networks. It is worth mentioning that, by allocating extra network resources to specific video flows or adjusting the forwarding behavior of switches, QoE-aware dynamic and demand-based resource allocation is seen as a method for increasing the user's perceived QoE in softwarized systems. To this end, we propose dynamic QoE-aware resource management in future softwarized networks to ensure that end-users receive streamed media services (SD and HD) with an improved and optimized QoE. In contrast to previous studies [37], [35], [44], the proposed strategy considers the dynamic link and switch resources management.

IV. A DYNAMIC QoE-AWARE RESOURCE MANAGEMENT ARCHITECTURE IN FUTURE SOFTWARIZED NETWORKS

This section provides a dynamic QoE-aware resource management architecture for future softwarized systems as indicated in Fig. 3. It consists of the following modules: QoE-sdnFlow monitor, QoE-sdnFlow manager, QoE monitoring and management, experience level agreements (ELA), database, network management, QoE contracts, QoE measurements and traffic route calculations and management. The QoE-sdnFlow monitor and QoE-sdnFlow manager modules were implemented as extensions of an existing POX controller. The ELA [51] component is integrated to provide a well-defined service level agreement (SLA), mechanism that is purely centred on the QoE of the end-users. The ELA also provides new pathways for QoE-aware business contracts between customers and SPs in future softwarized networks. It is worth noting that, the focus of ELA in the proposed architecture is to enable SPs, MNOs and OTTs to offer better quality of business (QoBiz) [5] and QoE-based multimedia streaming services in softwarized networks. A description of the QoE-aware resource management architecture components shown in Fig. 3 is given in the following sections.

A. QoE-SDN FLOW MONITOR

The QoE-sdnFlow monitor module is responsible for monitoring the overall utilization of the resources of mapped virtual links and nodes. It also monitors and manages the average load of the links and switches in the substrate network. In addition, the QoE-sdnFlow monitor performs QoE-estimation for multimedia traffic flows and QoE measurements. Based on the identified multimedia traffic type such as (e.g., video/voice), the QoE-sdnFlow monitor selects the QoE estimation/assessment model available for estimating QoE. The QoE-sdnFlow monitor generates QoE based

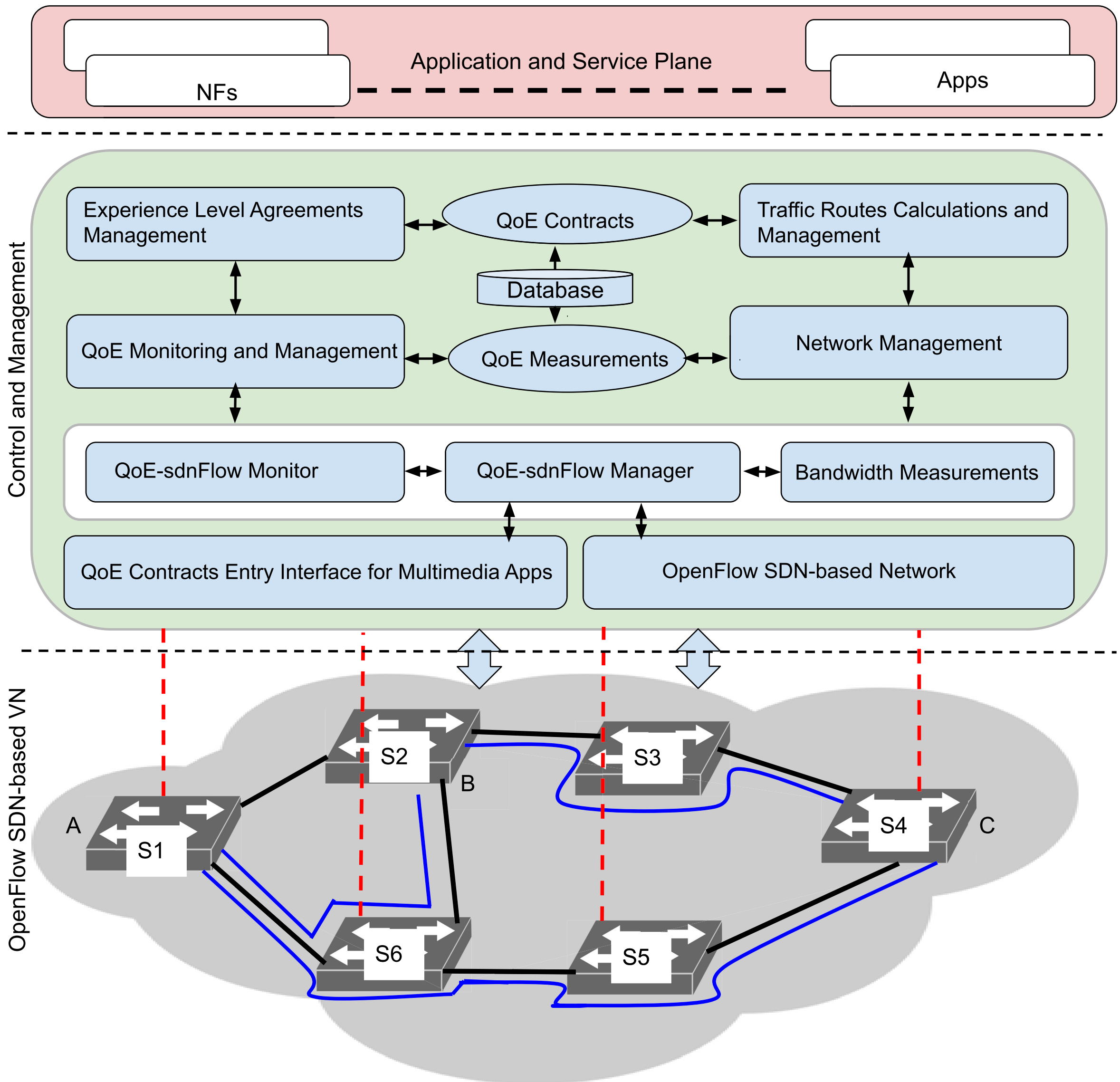


FIGURE 3. **Proposed QoE-aware dynamic resources management in future softwarized networks.**

inputs/parameters such as buffering events, the number of stalling events and their duration. It reports these inputs/data to the QoE-sdnFlow manager module. Furthermore, this entity/module performs multimedia traffic classification to understand the video content type of network traffic using statistical/parametric analysis. The per-client network resource requirements (QoS) are formulated and mapped to the optimal per-client QoE using the exponential mapping function indicated in Equation 1.

$$M_{function} = \begin{cases} QoS = k \times (W \times B_w) \\ QoE = \alpha \times e^{-}(\beta \times QoS) + \gamma. \end{cases} \quad (1)$$

W is a weighting parameter that shows the degree of QoS metrics for various services and applications in softwarized networks. For DASH services, the network bandwidth can represent the W of streaming services. B_w is the bandwidth, k indicates a constant factor for the access network and γ is a parameter for the video codec (e.g., VVC or H265). α and β indicate network level parameters that enable personalized QoS/QoE services (e.g., for gold, silver and bronze users) in softwarized networks.

B. QOESDNFLOW MANAGER

This module acquires information regarding the network topology after successful virtual network to substrate network mapping. The QoE-sdnFlow manager implements QoE-aware resource allocation and QoE based network policies/actions to manage the overall customer's multimedia streaming experience in softwarized networks. In addition, the QoE-sdnFlow manager can use different multimedia traffic control and management approaches for QoE-driven radio resource allocation, load balancing, QoE predictions, QoE-admission control and user density prediction in softwarized networks. That way, the QoE-sdnFlow manager for SPs, MNOs and OTTs enables them to enhance and optimize QoBiz with their customers via well-established QoE contracts interfaces for multimedia Apps. Notably, the QoE-sdnFlow manager is a state-action oriented component where a specific state in the network dictates an action/policy that should be taken while forwarding flow rules from the SDN switches.

1) ACTIONS ON FLOW RULES

An action is defined as an operation that is used to forward a video packet to an SDN switch port. The actions in a softwarized network are related to changing the rules of the switches. When new VNs are requested and old ones leave, the network state changes. These adjustments based on the action/operation that is used have an impact on the VNs' resource utilization and the loading of an SN. There are three possible actions, addition, modification and deletion which can be applied to forwarding flow rules in SDN switches of softwarized networks.

2) FLOW RULE ADDITION

When a VN is successfully mapped to an SN for all virtual links, the QoE-sdnFlow manager adds forwarding flow rules to the corresponding SDN switch. As shown in Fig. 3, for virtual link AB after successful mapping of VN ABC, the forwarding flow rules should be added by the controller to SDN switches $S1$, $S6$, $S2$ to establish the substrate path {S1→S6→S2}. The database stores these flow rules when the substrate path is established. This enables the past utilized resources and future available resources to be updated based on the user's demands.

3) FLOW RULE DELETION

When the VN departs or changes its assignment, the SN resources that has been allocated are released. Again, some forwarding flow rules are deleted by the QoE-sdnFlow manager module when a virtual link is migrated or departed. For example, a virtual link AB of a substrate path {S1→S6→S4} can be migrated to {S1→S4}. The flow rules in SDN switch B that are associated with the previous assignment are deleted because they become useless.

4) FLOW RULE MODIFICATION

The state of a network changes when old VNs leave and new ones are embedded. The modification of a flow rule on a substrate path within a softwarized network can be triggered by a departure or change in VN assignment. Virtual link AB can migrate from {S1→S6→S2} to {S1→S2} when network resources are available on the substrate link {S1→S2}. The forwarding flow rules in SDN switch $S1$ can be modified owing to this migration.

C. EXPERIENCE LEVEL AGREEMENT MANAGEMENT AND QoE CONTRACTS

The ELA formalizes QoE-oriented contracts between SPs and end-users. The proposed architecture shown in Fig. 3 integrates the QoE-oriented business model introduced in our work [5]. The simplified ELA ecosystem model in softwarized networks based on SLA and QoE concepts is shown in Fig. 4. The ELA ecosystem model provides QoE-related contracts between service providers and customers and ensure Quality of Business (QoBiz) in future softwarized systems. QoBiz quantifies business revenue from a service offered by a service/content provider on the Internet. The cost of servers, website revenue, transaction loss, and dollars per transaction are examples of QoBiz metrics expressed in monetary units [52]. The aim is to foster new business practices for ISPs such that multimedia streaming services can be delivered to the end-users with assured QoE. It is worth mentioning that MNOs, SPs and OTTs should provide QoE marketization to their customers using ELAs management strategies [5].

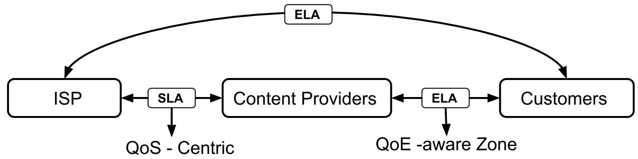


FIGURE 4. The ELA ecosystem in softwarized networks based on SLA and QoE concepts.

D. QoE MONITORING AND MANAGEMENT

This module can monitor the streaming status of clients and optimize the end-user's QoE of multimedia services. The input QoE parameters are monitored using specific settings as defined in the ELAs by QoE monitoring tools using an active or passive monitoring strategy. This component is responsible for collecting data from various parts of the network (e.g., radio access networks (RANs), user terminals) that may be used by ISPs/OTTs while delivering multimedia streaming services to users [43]. The proposed architecture is designed to have several monitoring points and a collaboration between service providers and network operators to: (a) provide real-time QoE monitoring assistance for 5G network users in residential, commercial and public locations, (b) provide insight and full access to E2E connection across all network segments, including: core networks, content server side networks, transit ISPs, aggregation networks, and customer premises networks, and (c) dynamically adjust QoE traffic flows for users (see Fig. 2) based on changing network conditions.

E. BANDWIDTH AND QoE MEASUREMENTS

The process of managing and optimizing end users' QoE requires an understanding of the core cause of QoE degradation or poor QoE levels. In this regard, important data and information on terminal capabilities (e.g., screen size and display performance), application/service, and QoE-related information inside the network must be monitored, gathered, and measured. This module is also used to measure the bandwidth of each virtual link and

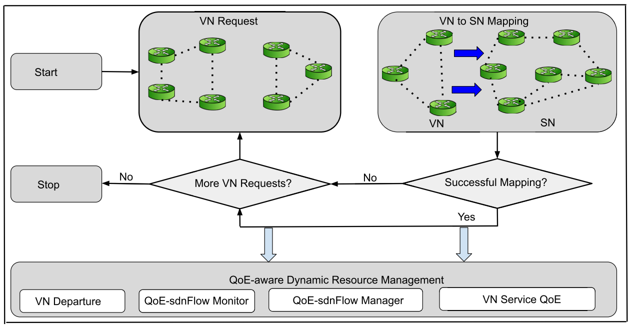


FIGURE 5. QoE-aware resource management algorithm in future softwarized network.

those that are migrated/remapped and require different resources/requirements.

F. DATABASE

The flow rules that are currently active during video streaming sessions are stored in a database. The SDN controller maintains the entry of active flow rules instead of querying the switches on a regular basis to construct tables across the southbound path. The flow rules of the mapped virtual links are added to the database and pushed to the impacted switches through the OpenFlow when VN mapping is successful. The mappings of virtual to substrate link and virtual to substrate switch are stored in the database. Moreover, the video streaming session link usage statistics, and resource availability are also stored in the database. It is worth noting that every time the flow rules are deleted from any of the SDN switches, they are also deleted from the database.

V. NETWORK MODEL AND QoE-AWARE RESOURCE MANAGEMENT ALGORITHM

A. NETWORK MODEL

We model the softwarized network as a weighted undirected graph $G_s = (V_s, E_s)$. E_s is a set of links and V_s indicates a set of SN switches. We used the graph $G_v^i = (V_v^i, E_v^i)$ to model a virtual network consisting of a set of virtual links E_v^i and virtual nodes V_v^i. Every SDN switch $(m \ldots n) \in S$ consists of a memory M_m, a TCAM [14] where the routing rules are stored. The substrate SDN links $l_{mn} \in L$ with B_w connects substrate switches m and n. The i_{th} VN assignment is performed using the link and node assignments as formulated below.

1) NODE ASSIGNMENT

Each virtual node is assigned to a substrate node formalized as a mapping $f_N^i : V_v^i \rightarrow V_s$ such that $f_N^i(v) \in V_s$, $\forall\, v \in V_v^i$ and $f_N^i(u) = f_N^i(v)$ iff $u = v$.

2) LINK ASSIGNMENT

The assignment of each virtual link to a substrate path between the corresponding substrate nodes is formalized using the following mapping:$f_L^i : E_v^i \rightarrow P_s$ such that $f_L^i(\overline{uv}) \in P_s(f_N^i(u), f_N^i(v))\ \forall\, (\overline{uv}) \in E_v^i$ where P_s is the set of all substrate paths. $\overline{uv}$ is the set of virtual links from the virtual nodes $\overline{u}$ to $\overline{v}$.

The VN departs when allocated SN resources are released. If an existing VN changes its assignment, the previously allocated resources are released by the SN. The released network resources are then allocated to a new assignment based on arriving VN requests. We define the $R_{u'v'}$ parameter as the total available resources utilization used by the virtual link $l_{u'v'}$ using Equation 2. $R_{u'v'}$ considers both the SDN switch memory $M_{u'}$ used by the forwarding flow rules associated with it and the network bandwidth $B_{u'v'}$.

$$R_{u'v'} = \alpha \sum_{l_{uv} \in (Pl_{u'v'})} (B_{u'v'}) + \beta \sum_{u \in (Pl_{u'v'})} (M_{u'}). \tag{2}$$

$B_{u'v'}$ is the network bandwidth used by the multimedia (video) flows of virtual link $l_{u'v'}$ during a video streaming session. $Pl_{u'v'}$ is the substrate path where the virtual link $l_{u'v'}$ is mapped. To determine the available network resources, $R_{u'v'}$ is evaluated every time a virtual network is mapped onto the substrate network. Equation 3 indicates the available substrate resources $A_{u'v'}$ referring to the total quantity of unallocated substrate link and switch resources. For each link $l_{u'v'}$, $A_{u'v'}$ is the sum of the resources over all switches and links on the substrate path $Pl_{u'v'}$. When it is time to make decisions about which virtual links to migrate, the substrate switches and links with limited resources are given priority.

$$A_{u'v'} = \alpha \sum_{l_{uv} \in (Pl_{u'v'})} (B_{uv} - B'_{uv}) + \beta \sum_{u \in (Pl_{u'v'})} (M_u - M'_u). \tag{3}$$

B'_{uv} is the network bandwidth of substrate link l_{uv} that has been allocated to virtual the links. M'_u is the total memory of the substrate switch u. M'_u is the memory allocated to the virtual SDN switches where it has been mapped. Equation 4 defines the weight $W_{u'v'}$ of the virtual links in softwarized networks.

$$W_{u'v'} = \lambda R_{u'v'} - \mu A_{u'v'}. \tag{4}$$

μ and λ are non-zero constants used to determine the availability of resources or resource utilisation on substrate links respectively. A high $W_{u'v'}$ value indicates that virtual link $l_{u'v'}$ consumes a significant amount of SN resources. This means that the SN is heavily loaded, $l_{u'v'}$ is inefficiently mapped and the remapping of such a virtual link must be considered. A low value of $W_{u'v'}$ signifies the underutilization of SN resources. It is worth mentioning that were are interested in giving preference to substrate links or switches with the fewest resources.

B. QoE-AWARE RESOURCE MANAGEMENT ALGORITHM

The QoE -aware dynamic resource management algorithm proposed in this study is indicated in Fig. 5. We specifically implement the QoE-aware sdnFlow monitor and QoE-aware sdnFlow manager as important components for QoE management and resource allocations in softwarized networks. One of the fundamental operations of network softwarization and virtualization is the assignment or mapping of VN resources to the SN. Each virtual link is assigned to a substrate path

TABLE 2. Summary of notations in problem analysis.

Symbol	Meaning
W	Weighting parameter
B_w	Network bandwidth
k	Indicates a constant factor for the access network
α and β	Network level parameters that enable personalized QoS/QoE services
E_s	A set of links
$R_{u'v'}$	Total available resources utilization used by the virtual link $l_{u'v'}$ using Equation 2
$B_{u'v'}$	Network bandwidth used by the multimedia (video) flows of virtual link $l_{u'v'}$
$Pl_{u'v'}$	Substrate path where the virtual link $l_{u'v'}$ is mapped
$A_{u'v'}$	Sum of the resources over all switches and links on the substrate path $Pl_{u'v'}$
Bf_{max}	Indicates the buffer size
$W_{u'v'}$	Weights of the virtual links
$P_{u'v'}$	Minimum acceptable price for streaming a video on a virtual link/path of length $l_{u'v'}$
μ^* and μ_s	Non-negative parameters that correspond to the rebuffering events and startup delay respectively

to which the corresponding end-points/nodes are connected. Again, each virtual node is assigned to one of the substrate nodes. As shown in Fig. 5, VNs consisting of different topologies and lifetimes are created based on VN demands/requests that arrive at any time in softwarized networks. When the VN request is received, the mapping of VN to SN is performed while ensuring that a balanced load on both the substrate nodes and links is achieved.

The dynamic QoE-sdnFlow manager module allocates the required resources to virtual links if the VN to SN mapping is successful. Other VN requests are considered if the VN to SN mapping is unsuccessful. The link re(mapping)/migrations can be performed by the QoE-sdnFlow manager in the network at any given VN service period if the state of the VN assignment is changed (e.g., new resources are required based on the user's demands). It is worth mentioning that, the proposed approach increases substrate resource utilization efficiency. In addition, it reduces network congestion and allows the SN to accommodate more VNs with limited resources. Whenever the state of the SN changes (e.g., a new VN request is accepted or a previously accepted request departs), the QoE-sdnFlow manager recalculates the weight $W_{u'v'}$ of each link $l_{u'v'}$ in the softwarized network. We summarize the QoE-aware dynamic resource management approach using Algorithm 1. Steps (12 - 17) indicate the beginning of a video streaming session, downloading a video segment and predicting the bandwidth for the next chunk to be downloaded.

C. VIDEO STREAMING AND QoE MAXIMIZATION MODEL IN SOFTWARIZED NETWORKS

Video streaming is modelled as a set of video chunks or segments. We formulate a video streaming QoE-driven quality optimization mechanism by assuming that, $N = \{1, 2, 3, \ldots, M\}$ segments lasting for t_s s and encoded at different bitrates are available in one video sequence for download during a streaming session. Let L be a set of all the available bitrate levels in a streaming session. The video player has the option of downloading segment n at a higher bitrate $L_n \in L$.

The DASH client requests video segments that satisfy their quality criteria based on the available bandwidth. The higher the bitrate, the better the user perceives video quality. The set of quality levels requested is $Q_r = \{Q1, Q2, \ldots, QN\}$ whereas the network bandwidths that need to download each video segment during a streaming session are $B_w = \{b1, b2, \ldots, bw\}$. The average video quality $V_{avg}(n)$ requested by a DASH client can be given by equation 5.

$$V_{avg}(n) = \frac{1}{N}\sum_{n=1}^{N} q(L_n). \tag{5}$$

We formulate the following key QoE elements for QoE optimization in softwarized networks during video streaming:

- **Startup Delay**: We assume that the startup delay is $T_s \leq Bf_{max}$. Bf_{max} is the buffer size which depends on the storage limitations of the video player and different service providers' policies. The buffer occupancy at time t is denoted by Bf(t) $\in [0, Bf_{max}]$. It is worth noting that the video segments are downloaded into a playback buffer that holds a video that has been downloaded but not yet viewed.
- **Average Video Quality Variations**: This indicates the magnitude of the variations in video quality from one segment to the next during a video streaming session. Equation 6 shows the average variations in video quality.

$$Vrations_{avg}(n) = \frac{1}{N-1}\sum_{n=1}^{N-1} |q(L_{n+1}) - q(L_{n-1})|. \tag{6}$$

- **Rebuffering Events**: During video streaming session, rebuffering occurs if the video download time exceeds the playout buffer level. The rebuffering time (R_t is given by Equation 7 below.

$$R_t(n) = \sum_{n=1}^{N}(\frac{d_t(L_n)}{d_s} - B_n). \tag{7}$$

- **Average Video Quality**: The average per-video quality for all downloaded segments is given by Equation 5

To efficiently provide video services delivery to end consumers while maintaining a high degree of QoE, we consider the aforementioned parameters in the video QoE calculations. These parameters (e.g., bitrate switching, rebuffering events, stalling, live latency etc.) which are key to video QoE are

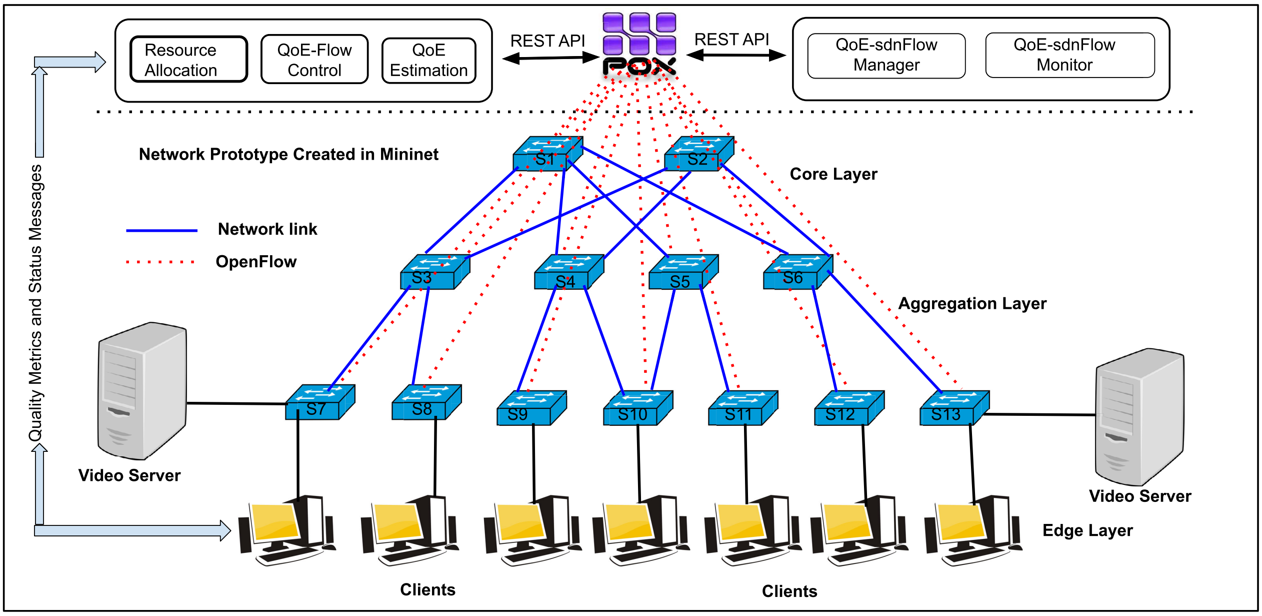


FIGURE 6. Experimental testbed for softwarized networks.

described in VI-E. The video QoE of the downloaded segments is calculated based on Equation 8 [53].

$$QoE_t^n = \frac{1}{N}\sum_{n=1}^{N} q(L_n) - W_{u'v'}\frac{1}{N-1}\sum_{n=1}^{N-1}|q(L_{n+1}) - q(L_{n-1})| - \mu^* \times \sum_{n=1}^{N}(\frac{d_t(L_n)}{d_s} - B_n) - \mu_s T_s\,. \quad (8)$$

$W_{u'v'}$ indicates the weights of the virtual links in a softwarized network as defined in Equation 4. A low value of $W_{u'v'}$ indicates that the SN resources are not fully utilized whereas a high value indicates that a significant amount of SN resources are utilized by the virtual link $l_{u'v'}$. μ^* and μ_s are non-negative parameters that correspond to the rebuffering events and startup delay respectively during a video streaming session. We set the values of μ^* and μ_s to 3000. This signifies that a 1s startup or rebuffering time receives the same penalty as reducing the bitrate of a video segment by 3000 kbps [53].

D. VIRTUAL NETWORK SURVIVABILITY AND QoE-BASED PRICING MODEL IN SOFTWARIZED NETWORKS

?? Substrate links and nodes in a virtual network can fail or be disrupted. The provision of resources for backups and/or restorations is an unavoidable aspect of any enduring resource management strategy in softwarized networks. To ensure that virtual nodes or links mapped to failing substrate resources are not affected, reactive or proactive virtual network embedding techniques can be employed. The proactive technique provides resources when the substrate network link/node fails whereas the reactive strategy provides backup resources before the failure of the substrate network link/node [54]. With reactive virtual network embedding, reserving some physical resources for unanticipated failures may result in wasteful substrate network resource utilization. The role of ISPs in a network virtualization environment (NVE) is

Algorithm 1 Dynamic QoE-aware Resource Management Algorithm

input : *NetworkTopology* $G = (V, E)$
Calculate link weight $W_{u'v'}$ *by using equation 4*;
Perform VN requests in the network;
Perform VN to SN mapping and assign each virtual link to a substrate path;
if *Mapping is successful* **then**
  Allocate the available substrate resources $A_{u'v'}$ *to each virtual* $l_{u'v'}$;
else
Perform more VN requests;
if *A VN changes its assignment* **then**
  Release SN previously allocated resources;
  Allocate new network resources based on the new assignment.;
end if;
Start a video streaming session;
for n = *1 to* N **do**
Calculate buffer size Bf_{max};
Calculate $V_{avg}(n)$, $Vrations_{avg}(n)$, T_s *and* $R_t(n)$ *using 5, 6 and 7*;
Download a video segment n with bitrate L_n ;
Perform prediction of bandwidth for the next chunk download by using a BandwidthPredictor function;
end for;
end if;

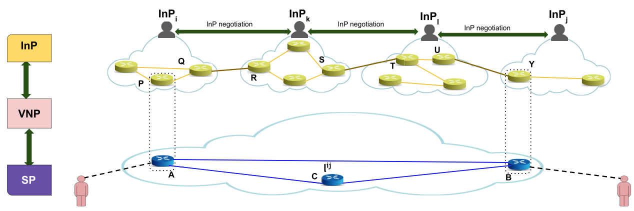


FIGURE 7. Multi-InPs collaboration/negotiation for survivable virtual softwarized networks.

divided into two categories: (a) service providers (SP) who implement different heterogeneous network architectures and network protocols on VNs composed of physical network resources that are leased from one or more IPs, and (b) IPs that deploy and maintain physical network resources such as links, routers and switches [54]. Survivability in NVEs from previous works [54], including our recent papers [36], [55] where we propose multipath protections and dynamic link recovery in softwarized 5G networks using segment routing has only been considered for a single InP environment.

In this study, we considered a collaboration/negotiation approach for multi-IPs as shown in Fig. 8. The aim is to optimize the InP's long-term revenue while minimizing the long-term penalty imposed by the InP because of QoS violations caused by failures. We considered a business scenario consisting of InPs and SPs where a virtual network provider (VNP) acts as a resource broker during service provision collaboration/negotiation. Assume that virtual link l_{ij} that has been mapped has to be provided with backup resources by the VNP. Note that nodes A and B have also been mapped by InPs InP_i and InP_j respectively. The same mapping procedure as that described in Section III-B and shown in Fig. 3 was used. During the provision of the backup resources, the set of InPs where the request can be sent is first determined. This includes InPs and neighboring InPs in the set that are involved in the mapping of the virtual link where resources need to be provisioned. Note that the VNP request includes the identity of the InPs that map each end of the virtual link. When the InPs receive a VNP request it determines whether it has enough substrate link resources to provide to the virtual link. If both ends of the virtual link are mapped within its domain, the InPs must additionally assess whether they can perform the mapping on their own. If the InP is capable of performing mapping on its own, it calculates the price of its resources using the pricing model shown in equation 9 and submits a proposal to the VNP. This hybrid pricing function shows the dynamic pricing scheme for a substrate network based on resource use. The pricing model minimizes network failures from overloading and encourages better network resource utilization. In contrast to the constant pricing models applied in various network virtualization works [56], the hybrid pricing model also ensures competitiveness for OTTs by ensuring that resources have reserve prices that cater to the minimum fixed costs of video streaming services to clients in softwarized networks [39], [57]. It is worth mentioning that the multi-InPs collaboration/negotiation substrate resource

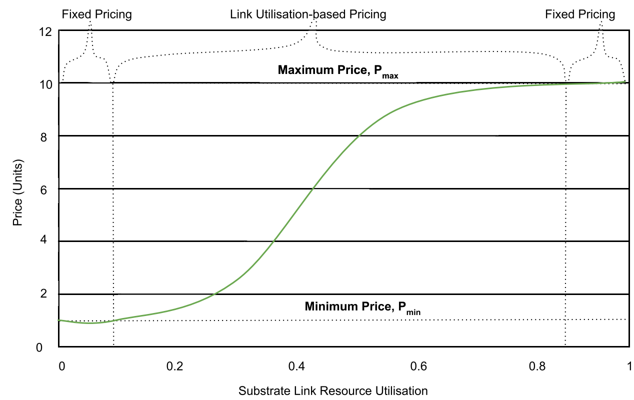


FIGURE 8. A dynamic multi-InPs collaboration/negotiation substrate resource QoS/QoE pricing in softwarized network.

QoS/QoE pricing is implemented in the testbed as indicated in Fig. 6.

The price per unit of transmitted video flow on a substrate link l_{mn} denoted as P_{mn} is given as shown in Fig. 8 and in equation 9 below.

$$P_{u'v'} = l_{u'v'}(P^{u'v'}_{min} + (\frac{P^{u'v'}_{max} - P^{u'v'}_{min}}{1 + \exp(c_1 - c2(R_{u'v'})})). \quad (9)$$

$P_{u'v'}$ is the minimum acceptable price for streaming a video on a virtual link/path of length $l_{u'v'}$. $R_{u'v'}$ is the resource utilization level and $P^{u'v'}_{max}$ is the maximum allowed price. c_1 and c_2 are constants that aim to shift the pricing function horizontally and determine its slope respectively. We also calculated the total price of the transmitted video flow from the OTT video server to the client over virtual link resources using Equation 10.

$$C_{mn} = \sum_{\forall W_{u'v'}} W_{u'v'} P_{u'v'}. \quad (10)$$

VI. PERFORMANCE ANALYSIS AND EVALUATION

A. EXPERIMENTAL TESTBED

We evaluated the performance and feasibility of the proposed architecture using a testbed as shown in Fig. 6 using a set of video streaming session-driven experiments. The innovativeness and credibility of our approach were also evaluated by extending the POX controller through the implementation of the QoE-sdnFlow monitor and QoE-sdnFlow manager components. These components perform QoE estimation and QoE optimization in softwarized networks to enhance the QoE of the end-user. They also monitor the overall utilization of resources of mapped virtual links and nodes and monitor or manage the average load of the links and switches of a substrate network. The QoE-sdnFlow manager, QoE-sdnFlow monitor, QoE estimation, QoE-Flow control, and resource allocation components communicate with the POX controller using the REST APIs. The communication of quality metrics and status messages from these components and video players is performed through REST APIs. The quality metrics and status messages exchanged include live latency, average link and switch resource utilization, video bitrate switching, startup delays, stall duration, average costs, income and profits. The SDN controller is integrated with the proposed QoE -aware dynamic resource management strategy presented in Algorithm 1 which implements the components/functions indicated in Fig. 6.

We employed BRITE [58] to generate topologies for VN requests in the network. We employ the top-down approach for generating hierarchical topologies in BRITE because we can interconnect the network nodes and assign topological components/attributes such as delay, bandwidth and packet loss. The SN was created in Mininet [59]. We performed virtual to SN mapping through node mapping using the greedy algorithm presented in [24]. We also use the multi-commodity flow (MCF) [60] approach to perform link mapping. The aim is to ensure that the video (commodities) flows through the network at a minimum cost without exceeding the arc bandwidth capacities. The bandwidth demand for virtual links configured on the SN as shown in Fig. 6 for edge, aggregation and core layers are 3 Mbps, 4 Mbps and 5 Mbps respectively.

It is important to mention that all messages include the required control messages in OpenFlow (e.g., (Link Layer Discovery Protocol (LLDP), Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), and other linkstate advertisement packets) that appear during the network uptime owing to the topology discovery in the SDN controller [61]. Again, the Spanning Tree Protocol (STP) is used in the topology indicated in Fig. 6, which adds more control messages to maintain the connections necessary for a loop-free tree network topology during video streaming [62]. This information and protocols were used in the analysis of SDN overhead using PACKET_IN and PACKET_OUT messages.

B. DASH.js: IMPLEMENTATIONS OVERVIEW

This section provides an overview of the dash.js reference player and the key components as shown in Fig. 9. While the dash.js player runs inside the web browser, video rendering is delegated to the media source extension (MSE) of the browser. We keep the standard-specific dash.js player implementations unchanged and make modifications and extensions to components, classes and functions related to video adaptive streaming. When a video streaming session starts from a video server to the end-user, the player loads a manifest (MPD) file that is parsed by the DASH Adapter to collect information (e.g., segment length, encoded bitrates, video length etc.) and possible live-streaming details such as live latency. Below, we describe the details of each class/function that plays a key role in video bitrate adaptation and video streaming.

- *ScheduleController*: The task of downloading a video segment or chunk is managed by the *ScheduleController* which also provides instructions to the *FragmentController* to download the segment or chunks at a certain video bitrate. It is important to note that the *ScheduleController* invokes *ABRController* to select a video bitrate from the server.
- *BufferController*: requests new video segments and manages the buffer levels of the dash.js video streaming player. The control and management of the buffer level enables the *BufferController* to make important decisions regarding bitrate changes during video streaming sessions. The validate function is invoked periodically every time a segment is downloaded and the *getPlaybackQuality* function from *AbrController* is called to retrieve an optimal video bitrate based on the available resources from the client's side. Buffer level variability is also maintained by the *BufferController* for recording the current streaming buffer occupancy of the dash.js player. This also helps to make better selection of the video bitrate.
- *ABRController*: The ABR rules are controlled and managed by *ABRController* which forms the core part of the video bitrate adaptation logic. *InsufficientBufferRule* selects the video bitrate based on the buffer level if it reaches the lower limit. *ThroughputRule* is a simple throughput-based algorithm that selects a video bitrate when the buffer levels are low (e.g., during video startup). *BolaRule* selects an appropriate bitrate corresponding to the estimated throughput when the buffer level is high. The *DroppedFramesRule* and *SwitchHistoryRule* are heuristic rules that avoid using video bitrates that exceed the computational resources of the device and extreme video bitrate oscillations allowed by the ABR algorithms respectively.

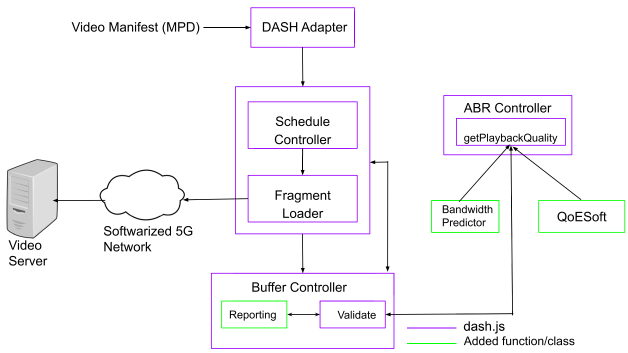
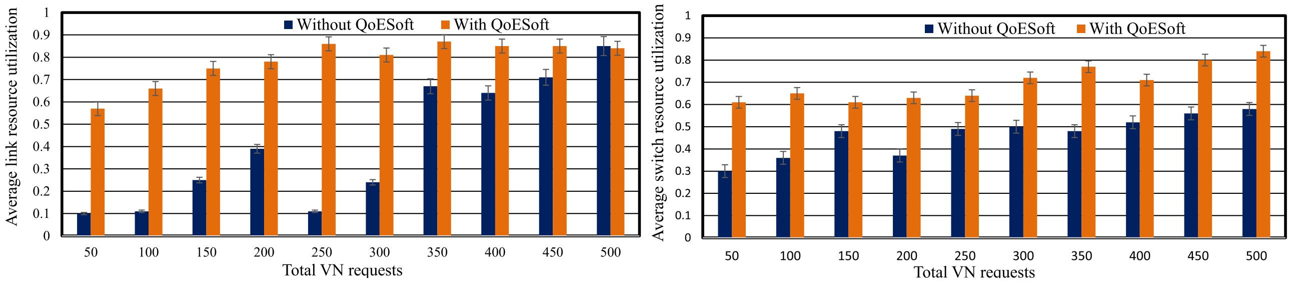


FIGURE 9. The simplified dash.js player code structure. Our modifications and extensions are incorporated in the ABR and Buffer controllers.

C. EXTENSIONS AND MODIFICATIONS

In addition to the modules (QoE-sdnFlow and QoE-sdnFlow) that we implemented in the POX controller, we made extensions and modifications to the *BufferController* and *ABRController* of the dash.js player [25]. We introduced a *reporting* function to provide the current streaming status of the dash.js player such as live latency, stall duration, selected bitrate, bitrate switches, normalized QoE, average link and switch resource utilization and startup delay. We also changed the bitrate decision adaptation logic and video segment download



(a) Average link resource utilization during video streaming

(b) Average switch resource utilization during video streaming

FIGURE 10. **Average link and switch resources utilization.**

FIGURE 11. **A SD and HD video clips from LIVE - Amazon Prime Video (APV) streaming database.**

process by ensuring that video bitrates are selected at the beginning of every segment downloaded. The *Bandwidth-Predictor* function ensures that the bitrate is selected based on the available resources on the client's side in a dynamic manner: it integrates a simple throughput-based algorithm that selects a video bitrate when the buffer levels are low (e.g., during video startup). *BandwidthPredictor* also selects an appropriate bitrate corresponding to the estimated throughput when the buffer level is high.

D. VIDEO PARAMETERS

The "Big Buck Bunny" video sequence with video resolution of 1920 × 1080 pixels, 9 minutes and 56 seconds was used for SD video transmission. Following the existing common media application format (CMAF)-based live services, we encoded SD and HD videos using FFmpeg (H.264/MPEG-4 AVC codec) version 3.3.4. The video sequence was segmented at (2, 4, 6, 8 and 10 seconds) using the GPAC MP4Box [63] for two different video resolutions (854 × 480)p for SD and 1280 × 720)p for HD with video encoding rates of 1.536Mbps and 2.496Mbps respectively. The video segments were then stored in the Apache server that is attached to the Mininet prototype running in a virtual machine as shown in Fig. 6. We executed the baseline dash.js player in a Google Chrome browser (v79) and considered 500 VN arrivals during video streaming and ran experiments 20 times from the server to the DASH client.

Furthermore, to provide different content on the video server, we used the LIVE - APV live streaming database[1]. The database consists of three reference videos. The reference videos were constructed from eight high -quality, uncom-

[1]https://live.ece.utexas.edu/research/Quality/TVSQ_VQA_database.html

TABLE 3. **List of used parameters.**

Video Type	Encoding rate (Mbps)	Frame rate (fps)	Resolution (p)
SD	1.536	30	480
HD	2.496	30	720

pressed video clips with different content. These video clips as shown in Fig. 11 had a spatial resolution of 720p (1280 × 720) for HD and a frame rate of 30 fps. For each reference video, five video sequence were segmented at (2, 4, 6, 8 and 10 seconds) using the GPAC MP4Box.

E. VIDEO PERFORMANCE METRICS

1) LIVE LATENCY

Live latency indicates the time taken from video capture to rendering at a point during the streaming session. This includes the time spent encoding a video. The dash.js player [25] provided real-time latency values.

2) AVERAGE LINK AND SWITCH RESOURCE UTILIZATION

We measured the efficient utilization of substrate resources in the SDN/NFV-based network by using two parameters: (a) average switch utilization, and (b) average link resource utilization.

- The average link resource utilization: Indicates the average proportion of the entire bandwidth capacity used by the substrate link at any given time.
- Average switch utilization: The average percentage of the total substrate switch capacity that is used at any one time.

3) QoE METRICS

We used the QoE model proposed by Yin et al. [53] to evaluate end-user QoE during video streaming. The QoE model combines four metrics: stall duration, selected bitrate, bitrate switches and startup delay.

- Video bitrate switching: Defines the number of times the video quality changes from one bitrate to another during transmission owing to a change in bandwidth or delays in the networks.
- Startup delays: This indicates the lag between a video object/frame chosen by a user and the actual playback starts on his/her screen. The time from when the viewer intended the video to play, to when the first frame of the

video was displayed on the user's device screen [5]. The startup delay also measures the time period before the first frame of the video can be displayed on the user's device screen.

- Stall duration: Measures the duration of stalling or rebuffering events that occurs during video streaming.

The video reception quality ρ [64] metric was also employed to evaluate the performance of video QoE. The ρ metric is the ratio between the download network throughput and video encoding rate (e.g., D_{thput}/V_{rate}). During the video streaming sessions, the ρ quantifies the impact of playback quality on the user's viewing experience. The video streaming viewing experience is good when the $\rho > 1$ otherwise it has a poor reception quality. The reception quality, average switch and link resource utilization, as well as other QoE metrics (e.g., selected bitrate, stall duration, startup delay, and bitrate switches) provided a complete performance analysis of the proposed strategy in comparison to the dash.js baseline approach.

F. EXPERIMENTAL RESULTS AND DISCUSSION

1) AVERAGE LINK RESOURCES UTILIZATION

The results for the average link resource utilization with and without QoESoft is indicated in Fig. 10a. The proposed approach, *QoESoft* outperforms the baseline approach in terms of link resources utilization because the resources are efficiently mapped based on the requirements of the end-user's VN requests in a softwarized network. The results indicate that as the number of VN requests increases the link utilization also increases. For example, when the VN requests are 300, QoESoft achieves more than 85% of resource utilizations while using the baseline approach is 25%. From 350 to 500 VN requests, the performance of the baseline approach is good, however, it performs less compared to the QoESoft approach. Fig. 10b shows the average switch resource utilization for 500 VN requests. Again, *QoESoft* achieves better results than the baseline approach used during video streaming. For example, at 300, 350 and 500 VN requests, QoESoft achieved 74%, 78% and 85% switch resource utilization respectively. When the baseline approach is used (e.g., without QoESoft) at these VN requests, only 50%, 48% and 58% of the link resources are utilized. The poor utilization of link and switch resources of the baseline approach is attributed to its static nature and inability to accept more VN requests because it can not perform node/link mapping. QoESoft performs better because of the dynamic nature of the SDN controller which that can accept more VN requests and allow the substrate network to effectively utilize its link/switch/node resources.

2) SELECTED BITRATE AND LIVE LATENCY DURING VIDEO STREAMING

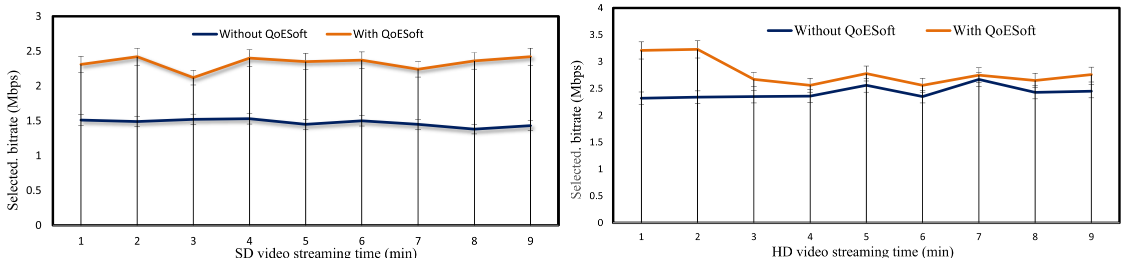
We provide a performance analysis of the selected bitrate and live latency during a streaming sessions. Fig. 12a and Fig. 12b indicate the selected bitrate for the two compared approaches (QoESoft and without QoESoft) for both SD and HD videos. The proposed strategy achieved the highest selected bitrate with an average of 2.33 Mbps for SD and 3.4 Mbps for HD video streaming. The average selected bitrate was 1.47 Mbps with a baseline approach (without QoESoft) for SD and 2.4 Mbps for HD videos. The proposed mechanism selects the video bitrate based on the available resources (bandwidth) during a streaming session through a BandwidthPredictor function (see Fig. 9) which integrates a throughput-based algorithm that selects a video bitrate when buffer level is low. The baseline approach performs poorly because it relies on the video adaptation logic which measures the bandwidth from the last downloaded segment and does not perform prediction. While not using QoESoft, the baseline strategy results into many bitrate switches (Fig.14, underutilization of network (link/switch) resources (Fig.10 leading to video instability and poor QoE as also reported by Barakabitze et al. [5].

Fig. 13a and Fig. 13b show the average live latency with and without QoESoft during an SD and HD video streaming session. QoESoft achieves a low average live latency for SD and HD video streaming compared to the baseline approach. It is worth noting that the proposed approach provides low latency and does not introduce a large number of stalls across all SD and HD video streaming sessions (see Fig. 13b). From these results we note that bandwidth prediction using *BandwidthPredictor* can greatly contribute to reducing latency during video streaming in future softwarized networks where 4K/8K and other ultra-high definition television (UHD) videos are expected to be transmitted with low latency and low power consumption. The higher latency without QoESoft is attributed to the dash.js that suffers from buffering events and a number of serial startup delays or stall durations most of which are proportional to the downloaded segment duration. QoESoft performs better because of the *BandwidthPredictor* function which accurately provides a shorter time available while reacting to bandwidth changes and video bitrate switching in softwarized networks. It is worth mentioning that QoESoft provides an average low latency of 1.74 sec for HD and 2.20 sec for SD videos which are acceptable low-latency (e.g., 2-10 sec) video streaming values for the OTT/ISPs, especially for streaming sports content/video [65].

3) THE NO. OF BITRATE SWITCHING DURING VIDEO STREAMING

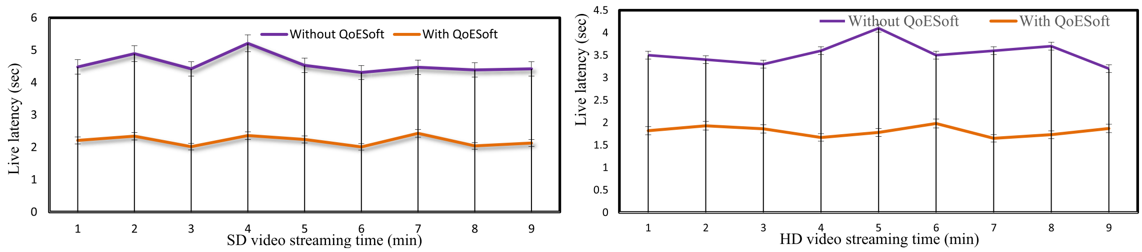
Fig. 14a and Fig. 14b show the performance results for the number of bitrate switches with and without QoESoft during SD and HD video streaming sessions. QoESoft achieves a maximum of 12 for SD videos and 7 for HD videos compared with the baseline approach which achieves a maximum of 23 for SD videos and 20 for HD videos.

The low bitrate switching values for QoESoft are attributed to the dynamic QoE resource management algorithm that efficiently utilizes the network resources (bandwidth) and



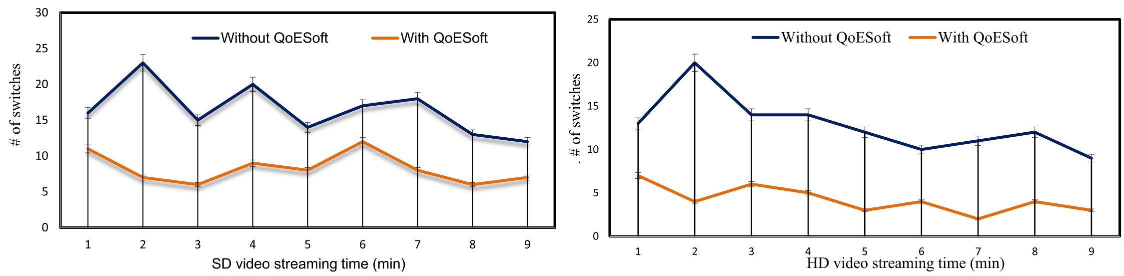
(a) Selected bitrate for SD videos (b) Selected bitrate for HD videos

FIGURE 12. The average selected video bitrate for SD and HD video streaming.



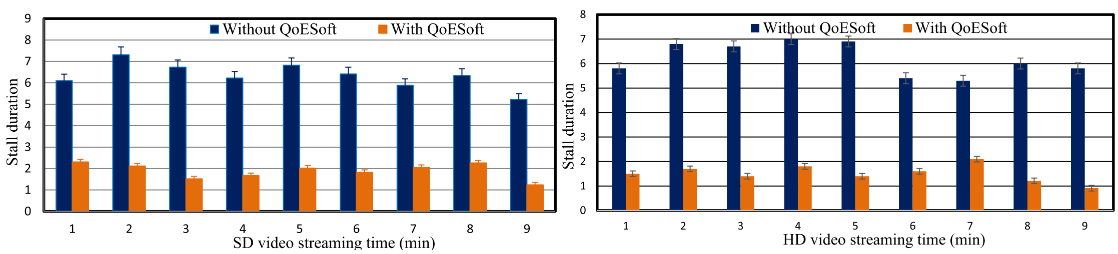
(a) Live latency during SD video streaming (b) Live latency during HD video streaming

FIGURE 13. The average live latency for SD and HD video streaming.



(a) Number of bitrate switching for SD videos (b) Number of bitrate switching for HD videos

FIGURE 14. The number of video bitrate switching for SD and HD video streaming.



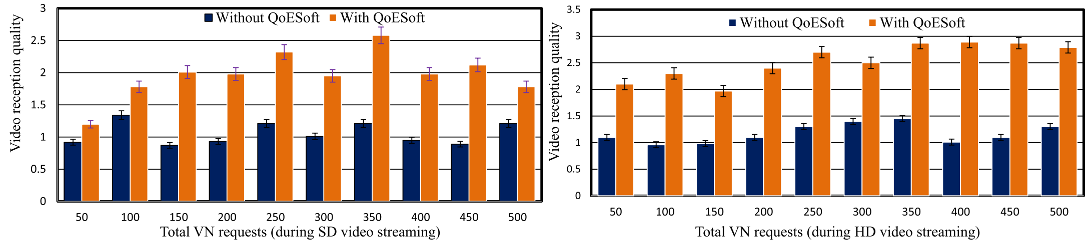
(a) Stall duration for SD video streaming (b) Stall duration for HD video streaming

FIGURE 15. The number of stall duration for SD and HD video streaming.

does not introduce over/undershoot bandwidth issues while achieving low-latency video streaming in softwarized networks. QoESoft avoids excessive video oscillations and an intermediate level is introduced through the *BandwidthPredictor* component in the switching process during SD and HD video streaming. As such, it ensures a gradual change (between quality changes) instead of switching suddenly up/down directly to the bitrate levels of the segment to be downloaded during video streaming such as how dash.js does [5], [25], [33].

4) STARTUP DELAY AND STALL DURATION DURING VIDEO STREAMING

The startup delays for SD and HD video streaming are indicated in Fig. 17 and Fig. 18. QoESoft achieves the best results for starting the streaming session delay achieving an average of 1.28 sec for SD videos and 1.29 sec for HD videos. This leads to a good QoE for end users during video streaming. This performance is attributed to the QoE-sdnFlow manager module which performs efficient resource allocation by selecting the optimal video representation while considering



(a) Reception quality of SD videos (b) Reception quality of HD videos

FIGURE 16. The streaming reception quality for SD and HD video streaming.

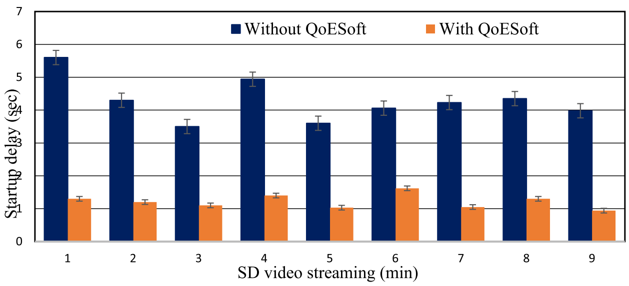


FIGURE 17. Startup delay of SD video streaming.

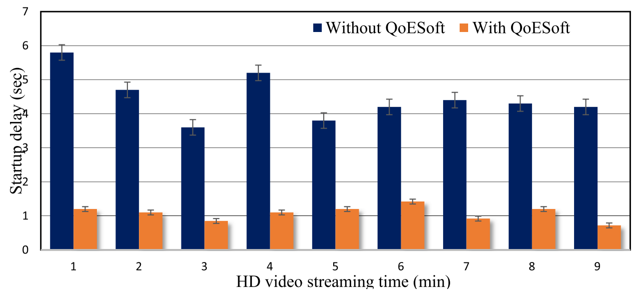


FIGURE 18. Startup delay for HD video streaming.

the changing network conditions and other client player -related factors. Moreover, sub-optimal decisions based on the QoE-sdnFlow manager have insight and full access to the E2E connection across all network segments as well as dynamic adjustment of QoE video traffic flows for users while benefitting from the QoE -aware dynamic resource management algorithm to ensure ensuring a fast startup delay. In addition, these results signify the low-live latency values presented in Fig. 13. The baseline approach performed poorly with an average of 4.35 sec for SD videos and 4.58 sec for HD videos. It is worth noting that higher startup delays without QoESoft are contributed by inefficient utilization of resources in the network and vice versa.

The performance results for stall duration are shown in Fig. 15a and Fig. 15b for SD and HD video streaming. The average stall duration with QoESoft approach for SD video streaming is 1.82 sec while without QoESoft is 5.87 sec. The average stall duration for HD video streaming is 1.51 sec and 6.59 sec with and without QoESoft respectively. The performance of QoESoft is good irrespective of the video definition when compared to the baseline dash.js approach, however, HD videos provide the best quality compared to that of SD videos. The overall performance results indicate that the proposed QoE resource management strategy (QoESoft) reduce bitrate, low-latency streaming, number of stalls and duration which leads to an improved QoE for the end-users.

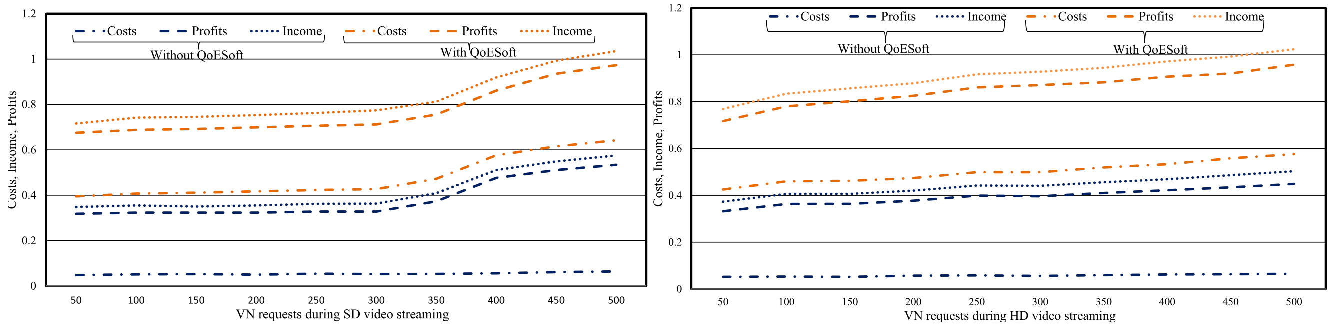
5) VIDEO QUALITY MEASUREMENTS

We also calculated video QoE using the QoE metric proposed in [53]. To demonstrate the performance of the proposed approach, we only used the 480p for SD video streaming and 720p for HD video resolution. Fig. 16 provides video streaming QoE measurement results for the SD and HD videos. The average video reception quality with QoESoft is 1.89 for SD videos and without QoESoft using a baseline approach is 0.93 for all VN requests (see. Fig. 16a). Fig. 16b indicates the average video streaming reception quality for HD of 1.11 without QoESoft and 2.23 with QoESoft. The video quality reception quality for QoESoft is good for every VN request compared to the other approach because of the efficient utilization of resources for both links and switches in softwarized networks.

Unsurprisingly, without using QoESoft, the video reception quality is good (>1) for the VN requests at 100, 250, 350 and 500. This is because network resources are utilized but at the cost of overhead at the SDN controller which lacks the allocation and reallocation of resources mechanism to the user's demands. Without QoESoft, the controller has no knowledge of monitoring and managing the average load of the virtual links and nodes of a substrate network leading to under/over-utilization of resources (bandwidth, link/nodes/switches) which are among the factors that contributes to poor video quality as stipulated in [5].

6) NORMALIZED QoE

Similar to a revenue model [33] that outputs a value between 0 and 1 to represent a player's satisfaction, we normalized the QoE value to [0,1] by dividing the online QoE achieved by the *QoESoft* algorithm to the maximum QoE that can be achieved with perfect knowledge of each bandwidth profile in the future. Fig. 20 shows the normalized QoE results for a 720p video resolution. The proposed approach achieves better performance than the baseline strategy. QoESoft achieves an average normalized QoE of 0.87 while without QoESoft is 0.66 for SD video streaming. QoESoft outputs an average noralized QoE of 0.90 and 0.69 for HD video streaming.



(a) Cost, income and profits for SD videos

(b) Cost, income and profits for HD videos

FIGURE 19. Cost, income and profits for SD and HD video streaming.

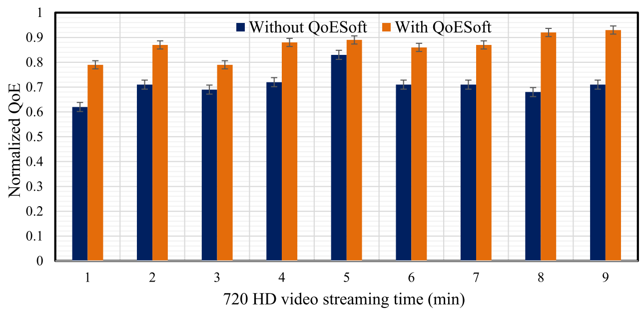


FIGURE 20. Normalized QoE for 720p video resolution.

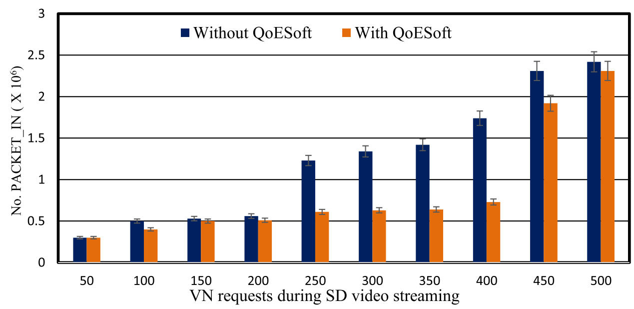


FIGURE 21. The number of PACKET_IN messages for each packet count in a SD video flow.

The significant improvement of the end-user's QoE using QoESoft is mainly attributed to the lower stall duration and startup delay, optimized video bitrate, lower live latency and lower number of bitrate switchings. The normalized QoE results for QoESoft are also supported by the video reception quality as indicated in Fig. 16. This is attributed by the dynamic QoEaware resource management mechanism where allocation and reallocation of resources (link and switch, as well as lower overhead of the SDN controller) are performed based on the user's requests in a softwarized network.

7) AVERAGE COSTS, INCOME AND PROFITS

We provide the performance results in terms of total costs (see Equation 10 based on the price per unit of a transmitted video flow using Equation 9. Note that for these results, we configured the mean time to repair (MTTR) and mean time between failures (MTBF) for each substrate link i a softwarized network. These configurations followed a Weibull distribution [66] and were both based on the characterization of link failures in a real ISP backbone performed in [67]. We are interested in these results because OTTs and ISPs are always interested in determining the prices of their resources (bandwidth, virtual network (link/switch) infrastructure) and QoE -based services from the transmitted videos they offer to customers. Figure 19 indicates that QoESoft performs slightly better in terms of the total costs, incomes and profits for transmitted SD and HD videos compared to the baseline approach. As the VN request increases, the total income and profits also increase for the SD and HD videos. This signifies that for the VN requested by customers or infrastructure users, there is a price paid to the OTTs/ISPs. The profits are even higher when customers are interested in streaming HD videos from OTTs than when streaming an SD video. ISPs are the entities most affected by the user churn, leading to a decrease in market share, business reputation and lower revenue/income as highlighted in [68] and [69]. QoESoft provides good economic benefits for OTTs/ISPs in terms of profit maximization by providing better QoE to the most profitable customers thereby reducing user churn.

8) SDN CONTROLLER OVERHEAD

An SDN controller experiences a load mainly when there is a PACKET_IN message (packets coming to the switch that do not match any flow rules inside an OF switch's flow table [70]. The packet pair of messages coming out is called the PACKET_OUT message, which is a packet containing flow modification (FLOW_MOD message) sent by the controller back to the switch [61]. Video flow packet circulation causes SDN controller overhead which deteriorates network performance in the context of SDN-based multimedia streaming because of excessive video flow of data. Reducing the number of PACKET_INs consequently reduces the number of PACKET_OUTs [70]. In the performance evaluation, this metric is evaluated by counting the number of PACKET_IN and PACKET_OUT messages during network up time for both SD and HD video streaming.

To evaluate the behavior of the SDN controller-switch communication overhead, we provide performance

TABLE 4. A summary of average performance QoE metrics for the baseline and QoESoft for SD.

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 Type	Reception Quality	Normalized QoE
Without QoESoft	4.57	6 & 6.3	0.89	4.28	16	95.40	SD	1.10	0.62
	4.58	4 & 5.6	1.3	4.34	15	96	SD	0.92	0.64
	4.55	6 & 5.7	2.92	4.42	14	97.5	SD	0.93	0.72
With QoESoft	2.20	3 & 1.89	2.33	1.22	8	89	SD	1.89	0.87
	2.24	3 & 1.74	2.35	1.25	5	87.3	SD	1.67	0.89
	2.21	4 & 1.82	2.42	1.32	5	90	SD	2.11	0.86

TABLE 5. A summary of average performance QoE metrics for the baseline and QoESoft for HD videos.

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 Type	Reception Quality	Normalized QoE
Without QoESoft	3.50	7 & 6.19	2.43	4.54	12	95.40	HD	1.17	0.70
	3.56	7 & 6.34	2.44	4.63	12	96	HD	1.11	0.67
	3.54	10 & 7.23	2.41	4.57	11	97.5	HD	1.23	0.72
With QoESoft	1.81	3 & 1.52	2.79	1.32	4	89	HD	2.34	0.87
	1.74	4 & 1.53	2.84	1.25	3	87.3	HD	2.21	0.89
	1.78	5 & 1.47	2.93	1.32	4	90	HD	2.15	0.94

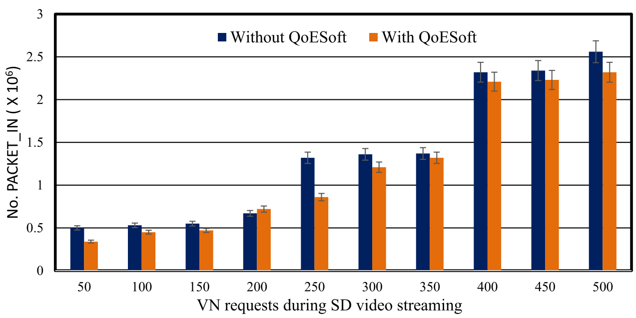


FIGURE 22. The number of PACKET_IN messages for each packet count in a HD video flow.

measurements for both PACKET_IN and PACKET_OUT message analysis. While the number of control messages is considered to be one of the root causes of high overhead in SDN controllers, our aim is to reduce the number of PACKET_IN message and PACKET_OUT messages during packet data of SD and HD video flow transmission which eventually reduces the SDN controller overhead. Fig. 21 and Fig. 22 indicate the number of PACKET_IN messages for each packet count in a SD and HD video flow respectively. QoESoft significantly reduces the controller overhead by achieving an average of 0.855×10^6 PACKET_IN messages for SD video flow transmissions compared to 1.24×10^6 PACKET_IN messages when the baseline approach is used.

In the baseline approach that employs the OpenFlow protocol, every switch involved in the forwarding path of a transmitted video should have flow rules installed by the SDN controller [6]. This requires significant resource consumption for caching at each switch and signaling overhead between switches and the controller for retransmission requests and caching releases [71]. This is different from QoESoft which involves only some switches in the forwarding path of a transmitted video to reduce use of available resources in an SDN-based network. SDN switch resources are less burdened with QoESoft owing to the reduced number of video flow entries. It is worth noting that fewer PACKET_IN messages are sent to the SDN controller during video flow transmission with QoESoft, which significantly reduces the amount of signaling exchange overhead.

9) SUMMARY OF RESULTS

Table(s) 4 and 5 summarize the performance results for the average QoE metrics for SD and HD videos. We note that QoESoft performs better than the baseline approach by avoiding many video bitrate switches while maintaining low video streaming latency. QoESoft also achieves a good viewing experience for both SD and HD videos, however, HD videos have a better performance than the dash.js baseline streaming. This performance is mainly attributed to the low startup delay, number of stalls and duration during video streaming sessions.

VII. CONCLUSION

Future networks such as 6G are anticipated to support and deliver high streaming multimedia QoE for a variety of applications and services (3D video, extended reality, 4K/8K/12K and video online gaming) to end-users. To achieve this objective, dynamic resource management mechanisms must be developed that can efficiently allocate resources to users based on their preferences. This paper presents a dynamic QoE-aware resource management method that considers both link and SDN switch resources in softwarized and virtualized networks. We proposed *QoESoft*, a network softwarization approach for multimedia streaming services that performs autonomic QoE resource management in future softwarized networks. We provide the initial evaluations of a virtual network survivability and QoE-based pricing models that maximize the profit for OTTs/ISPs by providing better QoE to the customers. Preliminary results indicate that *QoESoft* outperforms the baseline approach in terms of link resource utilization and switch resource utilization, low-live latency, startup delays, bitrate switching, stall duration and video QoE. Future work will extend this study and investigate QoE-aware dynamic resource management and control for multimedia streaming services while considering network slicing [72], video QoE encryption and scalability issues in 6G networks.

REFERENCES

[1] I. F. Akyildiz, A. Kak, and S. Nie, "6G and beyond: The future of wireless communications systems," *IEEE Access*, vol. 8, pp. 133995–134030, 2020.

[2] A. A. Barakabitze and R. Walshe, "SDN and NFV for QoE-driven multimedia services delivery: The road towards 6G and beyond networks," *Comput. Netw.*, vol. 214, Sep. 2022, Art. no. 109133.
[3] A. A. Barakabitze and A. Hines, *Multimedia QoE-Driven Services Delivery Toward 6G and Beyond Network*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 185–201.
[4] A. A. Barakabitze and A. Hines, *Multimedia Streaming Services Over the Internet*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 57–71.
[5] A. A. Barakabitze, N. Barman, A. Ahmad, S. Zadtootaghaj, L. Sun, M. G. Martini, and L. Atzori, "QoE management of multimedia streaming services in future networks: A tutorial and survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 526–565, 1st Quart., 2020.
[6] A. Alex, A. Ahmad, R. Mijumbi, and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Comput. Netw.*, vol. 167, pp. 1–40, Feb. 2020.
[7] 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," in *Proc. 8th Int. Conf. Quality Multimedia Exper. (QoMEX)*, Jun. 2016, pp. 1–2.
[8] A. A. Barakabitze and A. Hines, *Multimedia Streaming Services Delivery in 2030 and Beyond Networks*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 203–220.
[9] A. A. Barakabitze and A. Hines, *Emerging Applications and Services in Future 5G Networks*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 133–145.
[10] A. A. Barakabitze and A. Hines, *Network Softwarization and Virtualization in Future Networks: The promise of SDN, NFV, MEC, and Fog/Cloud Computing*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 99–118.
[11] A. A. Barakabitze and A. Hines, *Management of Multimedia Services in Emerging Architectures Using Big Data Analytics: MEC, ICN, and Fog/Cloud Computing*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 119–132.
[12] A. A. Barakabitze and A. Hines, *5G Networks*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 1–17.
[13] A. A. Barakabitze and A. Hines, *5G Network Management for Big Data Streaming Using Machine Learning*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 19–33.
[14] K. Kannan and S. Banerjee, "Compact TCAM: Flow entry compaction in TCAM for power aware SDN," in *Distributed Computing and Networking* (Lecture Notes in Computer Science), vol. 7730, D. Frey and M. Raynal, Eds. Berlin, Germany: Springer, Dec. 2013, pp. 439–444.
[15] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "A novel QoE-centric SDN-based multipath routing approach for multimedia services over 5G networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018, pp. 1–7.
[16] A. A. Barakabitze and A. Hines, *QoE Management of Multimedia Service Challenges in 5G Networks*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 167–183.
[17] W.-S. Lai, M.-E. Chiang, S.-C. Lee, and T.-S. Lee, "Game theoretic distributed dynamic resource allocation with interference avoidance in cognitive femtocell networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2013, pp. 3364–3369.
[18] T. Patikirikorala, A. Colman, J. Han, and L. Wang, "A multi-model framework to implement self-managing control systems for QoS management," in *Proc. 6th Int. Symp. Softw. Eng. Adapt. Self-Managing Syst.*, May 2011, pp. 218–227.
[19] W. Pan, D. Mu, H. Wu, and L. Yao, "Feedback control-based QoS guarantees in web application servers," in *Proc. 10th IEEE Int. Conf. High Perform. Comput. Commun.*, Sep. 2008, pp. 328–334.
[20] F. Jokhio, A. Ashraf, S. Lafond, I. Porres, and J. Lilius, "Prediction-based dynamic resource allocation for video transcoding in cloud computing," in *Proc. 21st Euromicro Int. Conf. Parallel, Distrib., Network-Based Process.*, Feb. 2013, pp. 254–261.
[21] A. A. Barakabitze and T. Xiaoheng, "Caching and data routing in information centric networking (ICN): The future internet perspective," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 4, pp. 26–36, Jan. 2014.
[22] A. A. Barakabitze, M. Liyanage, and A. Hines, "QoESoft: QoE management architecture for softwarized 5G networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–6.
[23] I. Fajjari, N. Aitsaadi, G. Pujolle, and H. Zimmermann, "VNR algorithm: A greedy approach for virtual networks reconfigurations," in *Proc. IEEE Global Telecommun. Conf.*, Dec. 2011, pp. 1–6.
[24] Y. Zhu and M. Ammar, "Algorithms for assigning substrate network resources to virtual network components," in *Proc. 25TH IEEE Int. Conf. Comput. Commun.*, Jun. 2006, pp. 1–12.
[25] (2019). *DASH Reference Player*. [Online]. Available: https://reference.dashif.org/dash.js/
[26] R. Farahani, F. Tashtarian, C. Timmerer, M. Ghanbar, and H. Hellwagner, "LEADER: A collaborative edge- and SDN-assisted framework for HTTP adaptive video streaming," in *Proc. IEEE Int. Conf. Commun.*, May 2022, pp. 745–750.
[27] R. Farahani, M. Shojafar, C. Timmerer, F. Tashtarian, M. Ghanbari, and H. Hellwagner, "ARARAT: A collaborative edge-assisted framework for HTTP adaptive video streaming," *IEEE Trans. Netw. Service Manage.*, vol. 20, no. 1, pp. 625–643, Mar. 2023.
[28] R. H. Majdabadi, M. Wang, and L. Rakai, "SODA-stream: SDN optimization for enhancing QoE in DASH streaming," in *Proc. IEEE/IFIP Netw. Operations Manage. Symp. (NOMS)*, Apr. 2022, pp. 1–5.
[29] A. Floris and A. Luigi, "Collaborative QoE management using SDN," *ACM SIGMultimedia Records*, vol. 12, p. 1, Dec. 2022.
[30] A. E. Al-Issa, A. Bentaleb, A. A. Barakabitze, T. Zinner, and B. Ghita, "Bandwidth prediction schemes for defining bitrate levels in SDN-enabled adaptive streaming," in *Proc. 15th Int. Conf. Netw. Service Manage. (CNSM)*, Oct. 2019, pp. 1–7.
[31] E. Liotou, K. Samdanis, E. Pateromichelakis, N. Passas, and L. Merakos, "QoE-SDN APP: A rate-guided QoE-aware SDN-APP for HTTP adaptive video streaming," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 598–615, Mar. 2018.
[32] R. I. Tavares da Costa Filho, W. Lautenschläger, N. Kagami, M. Caggiani Luizelli, V. Roesler, and L. Paschoal Gaspary, "Scalable QoE-aware path selection in SDN-based mobile networks," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 989–997.
[33] A. Bentaleb, A. C. Begen, R. Zimmermann, and S. Harous, "SDNHAS: An SDN-enabled architecture to optimize QoE in HTTP adaptive streaming," *IEEE Trans. Multimedia*, vol. 19, no. 10, pp. 2136–2151, Oct. 2017.
[34] A. Ahmad, A. Floris, and L. Atzori, "Timber: An SDN-based emulation platform for experimental research on video streaming," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 7, pp. 1374–1387, Jul. 2020.
[35] C. Benzaid and T. Taleb, "AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions," *IEEE Netw.*, vol. 34, no. 2, pp. 186–194, Mar. 2020.
[36] A. Barakabitze, I.-H. Mkwawa, A. Hines, L. Sun, and I. Emmanuel, "QoEMultiSDN: Management of multimedia services using MPTCP/SR in softwarized and virtualized networks," *IEEE Commun. Surveys Tuts.*, vol. 1, pp. 526–565, 2019.
[37] S. Sun, L. Gong, B. Rong, and K. Lu, "An intelligent SDN framework for 5G heterogeneous networks," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 142–147, Nov. 2015.
[38] R. Mijumbi, J.-L. Gorricho, J. Serrat, M. Claeys, F. De Turck, and S. Latré, "Design and evaluation of learning algorithms for dynamic resource management in virtual networks," in *Proc. IEEE Netw. Operations Manage. Symp. (NOMS)*, May 2014, pp. 1–9.
[39] R. Mijumbi, S. Hasija, S. Davy, A. Davy, B. Jennings, and R. Boutaba, "Topology-aware prediction of virtual network function resource requirements," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 1, pp. 106–120, Mar. 2017.
[40] R. Mijumbi, S. Hasija, S. Davy, A. Davy, B. Jennings, and R. Boutaba, "A connectionist approach to dynamic resource management for virtualised network functions," in *Proc. 12th Int. Conf. Netw. Service Manage. (CNSM)*, Oct. 2016, pp. 1–9.
[41] E. Grigoriou, A. A. Barakabitze, L. Atzori, L. Sun, and V. Pilloni, "An SDN-approach for QoE management of multimedia services using resource allocation," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Paris, France, May 2017, pp. 1–7.
[42] A. A. Barakabitze and A. Hines, *Quality of Experience Management of Multimedia Streaming Services*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 1–17.
[43] A. A. Barakabitze and A. Hines, *QoE Management of Multimedia Services Using Machine Learning in SDN/NFV 5G Networks*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 73–97.
[44] C. Tselios and G. Tsolis, "On QoE-awareness through virtualized probes in 5G networks," in *Proc. IEEE 21st Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD)*, Oct. 2016, pp. 159–164.

[45] A. Erfanian, F. Tashtarian, A. Zabrovskiy, C. Timmerer, and H. Hellwagner, "OSCAR: On optimizing resource utilization in live video streaming," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 1, pp. 552–569, Mar. 2021.

[46] R. Mijumbi, J. Serrat, J. Rubio-Loyola, N. Bouten, F. De Turck, and S. Latré, "Dynamic resource management in SDN-based virtualized networks," in *Proc. 10th Int. Conf. Netw. Service Manage. (CNSM) Workshop*, Nov. 2014, pp. 412–417.

[47] T. Feng, J. Bi, and K. Wang, "Joint allocation and scheduling of network resource for multiple control applications in SDN," in *Proc. IEEE Netw. Operations Manage. Symp. (NOMS)*, May 2014, pp. 1–7.

[48] A. S. D. Alfoudi, S. H. S. Newaz, A. Otebolaku, G. M. Lee, and R. Pereira, "An efficient resource management mechanism for network slicing in a LTE network," *IEEE Access*, vol. 7, pp. 89441–89457, 2019.

[49] A. Martin, J. Egaña, J. Flórez, J. Montalbán, I. G. Olaizola, M. Quartulli, R. Viola, and M. Zorrilla, "Network resource allocation system for QoE-aware delivery of media services in 5G networks," *IEEE Trans. Broadcast.*, vol. 64, no. 2, pp. 561–574, Jun. 2018.

[50] P. Zhou, Y. Xie, B. Niu, L. Pu, Z. Xu, H. Jiang, and H. Huang, "QoE-aware 3D video streaming via deep reinforcement learning in software defined networking enabled mobile edge computing," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 1, pp. 419–433, Jan. 2021.

[51] M. Varela, P. Zwickl, P. Reichl, M. Xie, and H. Schulzrinne, "From service level agreements (SLA) to experience level agreements (ELA): The challenges of selling QoE to the user," in *Proc. IEEE Int. Conf. Commun. Workshop (ICCW)*, Jun. 2015, pp. 1741–1746.

[52] D. Rivera, N. Kushik, C. Fuenzalida, A. Cavalli, and N. Yevtushenko, "QoE evaluation based on QoS and QoBiz parameters applied to an OTT service," in *Proc. IEEE Int. Conf. Web Services*, Jun. 2015, pp. 607–614.

[53] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," in *Proc. ACM Conf. Special Interest Group Data Commun.*, Aug. 2015, pp. 325–338.

[54] N. Shahriar and R. Boutaba, "Survivable virtual network embedding," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, May 2021, pp. 748–753.

[55] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "Multipath protections and dynamic link recoveryin softwarized 5G networks using segment routing," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2019, pp. 1–6.

[56] M. R. Rahman and R. Boutaba, "SVNE: Survivable virtual network embedding algorithms for network virtualization," *IEEE Trans. Netw. Service Manage.*, vol. 10, no. 2, pp. 105–118, Jun. 2013.

[57] R. Mijumbi, J. Serrat, and J.-L. Gorricho, "Self-managed resources in network virtualisation environments," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, May 2015, pp. 1099–1106.

[58] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An approach to universal topology generation," in *Proc. 9th Int. Symp. Model., Anal. Simul. Comput. Telecommun. Syst.*, 2001, pp. 346–353.

[59] R. L. S. de Oliveira, C. M. Schweitzer, A. A. Shinoda, and L. R. Prete, "Using mininet for emulation and prototyping software-defined networks," in *Proc. IEEE Colombian Conf. Commun. Comput. (COLCOM)*, Jun. 2014, pp. 1–6.

[60] C. Barnhart, "Multicommodity flow problems," in *Encyclopedia of Optimization*, C. A. Floudas and P. M. Pardalos, Eds. Phuket, Thailand: IEEE, Oct. 2009, pp. 2354–2362.

[61] C. S. Khin, M. Zin Oo, and A. T. Kyaw, "Packet-in messages handling scheme to reduce controller bottlenecks in OpenFlow networks," in *Proc. 17th Int. Conf. Electr. Engineering/Electronics, Comput., Telecommun. Inf. Technol. (ECTI-CON)*, Jun. 2020, pp. 502–505.

[62] (2014). *Control Messages*. [Online]. Available: https://techhub.hpe.com/eginfolib/networking/docs/sdn/sdnc2_4/5998-6899prog/content/c_sdnc-OpenFlow-subjects.html

[63] (Dec. 19, 2022). *GPAC: Multimedia Open Source Project*. [Online]. Available: https://gpac.wp.imt.fr/mp4box/

[64] T. Hofeld, R. Schatz, E. Biersack, and L. Plissonneau, "Internet video delivery in YouTube: From traffic measurements to quality of experience," *Data Traffic Monitoring and Analysis*, vol. 7730. Cham, Switzerland: Springer, Jun. 2013, pp. 264–301.

[65] A. Bentaleb, A. C. Begen, S. Harous, and R. Zimmermann, "Data-driven bandwidth prediction models and automated model selection for low latency," *IEEE Trans. Multimedia*, vol. 23, pp. 2588–2601, 2021.

[66] R. B. Abernethy, *The New Weibull Handbook: Reliability and Statistical Analysis for Predicting Life, Safety, Supportability, Risk, Cost and Warranty Claims*, 5th ed. IEEE/ACM Transactions on Networking, Nov. 2006, pp. 1–360.

[67] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, Y. Ganjali, C. Diot, and C. Diot, "Characterization of failures in an operational IP backbone network," *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 749–762, Aug. 2008.

[68] A. Floris, A. Ahmad, and L. Atzori, "QoE-aware OTT-ISP collaboration in service management: Architecture and approaches," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 14, no. 2s, pp. 1–24, Apr. 2018.

[69] A. Ahmad and L. Atzori, "MNO-OTT collaborative video streaming in 5G: The zero-rated QoE approach for quality and resource management," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 1, pp. 361–374, Mar. 2020.

[70] A. A. Pranata, T. S. Jun, and D. S. Kim, "Overhead reduction scheme for SDN-based data center networks," *Comput. Standards Interfaces*, vol. 63, pp. 1–15, Mar. 2019.

[71] M. Obadia, M. Bouet, J.-L. Rougier, and L. Iannone, "A greedy approach for minimizing SDN control overhead," in *Proc. 1st IEEE Conf. Netw. Softwarization (NetSoft)*, Apr. 2015, pp. 1–5.

[72] A. A. Barakabitze and A. Hines, *5G Network Slicing Management Architectures and Implementations for Multimedia*. Hoboken, NJ, USA: Wiley-IEEE Press, Dec. 2023, pp. 147–167.