



This is a peer-reviewed, final published version of the following document, © 2022 by the authors. Licensee MDPI, Basel, Switzerland. and is licensed under Creative Commons: Attribution 4.0 license:

**Al Jameel, Mohammed, Kanakis, Triantafyllos, Turner, Scott, Al-Sherbaz, Ali ORCID logoORCID: <https://orcid.org/0000-0002-0995-1262> and Bhaya, Wesam S (2022) A Reinforcement Learning-Based Routing for Real-Time Multimedia Traffic Transmission over Software-Defined Networking. Electronics, 11 (15). doi:10.3390/electronics11152441**

Official URL: <https://doi.org/10.3390/electronics11152441>

DOI: <http://dx.doi.org/10.3390/electronics11152441>

EPrint URI: <https://eprints.glos.ac.uk/id/eprint/11503>

#### **Disclaimer**

The University of Gloucestershire has obtained warranties from all depositors as to their title in the material deposited and as to their right to deposit such material.

The University of Gloucestershire makes no representation or warranties of commercial utility, title, or fitness for a particular purpose or any other warranty, express or implied in respect of any material deposited.

The University of Gloucestershire makes no representation that the use of the materials will not infringe any patent, copyright, trademark or other property or proprietary rights.

The University of Gloucestershire accepts no liability for any infringement of intellectual property rights in any material deposited but will remove such material from public view pending investigation in the event of an allegation of any such infringement.

PLEASE SCROLL DOWN FOR TEXT.

## Article

# A Reinforcement Learning-Based Routing for Real-Time Multimedia Traffic Transmission over Software-Defined Networking

Mohammed Al Jameel <sup>1,2,\*</sup>, Triantafyllos Kanakis <sup>1</sup>, Scott Turner <sup>3</sup>, Ali Al-Sherbaz <sup>4</sup> and Wesam S. Bhaya <sup>2</sup>

<sup>1</sup> Department of Computing, University of Northampton, Northampton NN1 5PH, UK

<sup>2</sup> Department of Computer Network, College of Information Technology, University of Babylon, Babil 51001, Iraq

<sup>3</sup> School of Computing, Canterbury Christ Church University, Canterbury CT1 1QU, UK

<sup>4</sup> Department of Technical and Applied Computing, University of Gloucestershire, The Park, Cheltenham GL50 2RH, UK

\* Correspondence: mohammed.aljameel@northampton.ac.uk

**Abstract:** Recently, video streaming services consumption has grown massively and is foreseen to increase even more in the future. The tremendous traffic usage has negatively impacted the network's quality of service due to network congestion and end-to-end customers' satisfaction represented by the quality of experience, especially during evening peak hours. This paper introduces an intelligent multimedia framework that aims to optimise the network's quality of service and users' quality of experience by taking into account the integration of Software-Defined Networking and Reinforcement Learning, which enables exploring, learning, and exploiting potential paths for video streaming flows. Moreover, an objective study was conducted to assess video streaming for various realistic network environments and under low and high traffic loads to obtain two quality of experience metrics; video multimethod assessment fusion and structural similarity index measure. The experimental results validate the effectiveness of the proposed solution strategy, which demonstrated better viewing quality by achieving better customers' quality of experience, higher throughput and lower data loss compared with the currently existing solutions.

**Keywords:** video streaming services; QoE; QoS; SDN; reinforcement learning



**Citation:** Al Jameel, A.; Kanakis, T.; Turner, S.; Al-Sherbaz, A.; Bhaya, W.S. A Reinforcement Learning-Based Routing for Real-Time Multimedia Traffic Transmission over Software-Defined Networking. *Electronics* **2022**, *11*, 2441. <https://doi.org/10.3390/electronics11152441>

Academic Editor: Martin Reisslein

Received: 7 July 2022

Accepted: 26 July 2022

Published: 5 August 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The unparalleled rise in video streaming traffic due to COVID-19 restrictions, touching almost 60% of overall bandwidth, and the strict Quality of Service (QoS) provisions of different applications impose great tension on the underlying network infrastructure [1]. Therefore, users' Quality of Experience (QoE) provisioning has become a critical challenge even faced by 5G/future networks. Due to network resource constraints and an enormous scope of applications, it would be challenging to guarantee different QoS requirements and thus significantly impact the users' perceived QoE [2]. As a result, QoS provisioning support has evolved into a required field in this research scope, particularly for services and applications that need to transfer data under specific QoS requirements (e.g., multimedia applications, video conferencing, video games, etc.).

Video streaming services from an extended selection of different platforms are rapidly taking the lead in most data-hungry applications in the digital world. Their demands are directly proportional to the video content resolution. In that event, the bandwidth is vital for video streaming and the fact that a high video resolution indicates a high bitrate. Thus the network has to accommodate more volume. Accordingly, giant services, including YouTube and Netflix, agreed to degrade their default streaming resolution to standard definition during the recent pandemic in order to conserve bandwidth. Although that

did not prohibit subscribers from adjusting the streaming resolution, it affected the end-user QoE [1]. Further, inadequate bandwidth increases the delay and loss rate, which leads to decreased end-user QoE. Video streaming is more vulnerable to data loss with high encoding bitrates since they severely impact the video streaming QoE. Users could confront frame freezing, complete video loss, or other problems depending on the lost video frames. On this basis, new and innovative technology solutions are investigated to support the high traffic requirements and provide better QoS and QoE, including Software Defined Networks (SDN) and Network Function Virtualisation (NFV) [3–5] and Artificial Intelligence (AI) [6–8].

SDN is identified as one of the key enabling technologies for modern networks, which provides numerous benefits to improve networks in relation to a logically centralised control model, network programmability, global view and the detachment of the control plane and data plane. The features of the SDN-based environment are suited for the deployment of multimedia bandwidth-hungry applications, such as video streaming [9]. The softwarised networks can be seen as a promising field for enhancing network performance which recently has brought both academic institutions and industry to study further issues in network performance optimisation. Additionally, AI and its subset machine learning (ML) integration has acquired an increasing reputation due to its utilisation in practically every sector [10–13].

This article introduces a new approach of Reinforcement Learning (RL)-based multimedia traffic transmission in SDN environment that considers QoS metrics and network information to enable exploring, learning, and exploiting potential paths for video streaming traffic. The proposed approach aims to enhance the satisfaction degree of end-users represented by QoE, towards Dynamic Adaptive Streaming over HTTP (DASH) based video flows leveraging the RL in SDN. It is implemented using emulation with real video content streamed by a multimedia provider and evaluated against other existing solutions, under different realistic SDN-enabled networks in terms of bandwidth, latency, jitter, loss, structural similarity index measure (SSIM) and video multimethod assessment fusion (VMAF).

The remainder of this article is organised as follows. Section 2 brings up some existing works on adaptive routing for multimedia traffic over SDN-enabled networks. Section 3 illustrates the proposed RL-based multimedia traffic routing framework. Section 4 highlights the problem domain and the proposed RL-based decision making solution. Section 5 presents the evaluation of the proposed approach. Section 6 describes the experiments' results that back our proposed RL-based solution. Finally, conclusion and future work are presented in Section 7.

## 2. Related Work

SDN draws a notable progression in networking technology and attracts attention from many researchers to investigate its use and address further its advantages [14]. It is worth noting that dynamic traffic routing approaches support different standards and can dynamically manage traffic flows by observing the status of the network and flow state. Network traffic flows can take any of the accessible paths with adaptable bandwidth based on various goals. A dynamic approach of QoS routing for video streaming over SDN-enabled networks was carried out in [15]. They presented an analytical framework for the optimisation of forwarding decisions for adaptive video streaming. Video streaming was given the highest routing priority, while the remaining services were granted as best-effort flows. Then, their routing approach was modeled into a Constrained Shortest Path (CSP) problem, which can be solved by employing the Lagrange Relaxation based Aggregated Cost (LARAC) algorithm [16]. Research in [17] utilised SDN environment and introduced an adaptive routing method, ARVS, for video streaming that supports QoS. They split video streaming into two layers to be used as a two-levels of QoS flows. They took delay variation as a given constraint to the CSP problem. If the jitter constraint is not satisfied, the video is re-routed to another available path on the base layer, and the

video enhancement layers remain on the same shortest path. Their method can enhance the quality of streaming of scalable encoded videos. Ongaro et al. [18] presented Integer Linear Programming (ILP) methodology for the QoS and QoE optimisation challenges in SDN networks concerning loss rate and delay. They also counted on network constraints and real-time service demands, such as maximum permitted loss and delay rates.

In [19] developers illustrated the impact of the Open Shortest Path First (OSPF) protocol over SDN-enabled networks. The work evaluated the resilience of network factors in SDN and traditional networks. Round trip time, convergence time and the QoS have been taken as the network parameters during the video streaming. The results indicated that applying OSPF in SDN networks gives less performance compared to its performance in classic networks; however, there is an improvement in QoS performance. Another work [20] benefited from the integration of ML with SDN to authorise a dynamic computation of the routing metrics for multimedia traffic. They proposed a modification in the OSPF protocol formula concerning the network QoS factors: bandwidth, delay, and loss rate. Also, a protocol for exchanging messages was run between the SDN controller and nodes in order to adjust the link-state metrics based on the present topology state. The outcomes indicate that the bandwidth utilisation increased, and the delay and packet loss rate decreased in multimedia traffic flows. Authors in [21] presented a strategy of flow-based video streaming routing over SDN. This strategy emphasised satisfying two QoS metrics, packet loss ratio and bandwidth. It selects reliable paths for each video streaming flow based on the existing state of the entire network. The experiment results reveal that HD videos are affected more than SD videos in the event of more packet loss occurring since they acquire more frame artefacts and colour distortions.

Concerning utilising ML methods with SDN, the work in [22] presented a new RL method to determine the optimal time for modifying the video bit rate and re-routing the traffic flows in order to minimise the data loss rate. They stated that their approach outperforms the traditional routing and greedy-based approaches. However, only small-scale topology scenarios are discussed in this study. Following this pioneered, Sendra et al. [23] introduced a routing optimisation strategy in SDN by employing the RL technique to improve network QoS. Their approach used the RL agent to choose the optimal paths that obtain the lowest cost by considering three parameters: delay, packet loss rate, and bandwidth. Authors in [24] proposed an intelligent QoS management framework for video traffic over SDN named LearnQoS. Their framework uses RL to improve the operation of a policy-based network management to guarantee the compliance of QoS demands for multimedia traffic over SDN. Besides, the work in [8] presented an adaptive approach for controlling multimedia traffic flow in an SDN-enabled network with a deep reinforcement learning method. Their system can realise the flow control policy directly from experience and assign bandwidth in order to maximise its total reward represented by customers' QoE. Another technique to utilise RL has been presented in [25] to satisfy the QoS demands. They proposed an RL-driven QoS-aware routing algorithm. Authors consider a QoS monitoring module to compute delay and packet-loss ratio and RL-based intelligent routing decision-making (RIRD) to interact with the monitoring system and obtain the best path. When the RL agent chooses the route with the minimum delay and packet-loss ratio during implementation, it should receive the highest reward value. Authors in [26] employed a Q-learning as a congestion-aware routing protocol over SDN called (QCAR). They utilised a set of parameters to denote the node and link status, node queue length, the hop count to the destination and re-transmitted packet rate. These parameters were measured periodically and used by the RL agent to calculate the Q-value in order to obtain the best path by avoiding congestion. The results indicate that QCAR performed better than the existing methods and showed an improvement of more than 15%. Jawad et al. [27,28] presented a new RL-based framework that determines the most suitable algorithm from a set of state-of-the-art routing algorithms to be used on the QoS-based traffic flows for the sake of QoS provisioning improvement. Guo et al. [29] focuses on traffic engineering in hybrid SDNs. It presents the RL method to achieve link load balancing while avoiding routing iterations by reacting



to the dynamic change in network traffic. Moreover, in [30], authors employed a deep reinforcement learning (DRL) technique for routing optimisation over SDN. The network topology operated in the data plane was data-centre networks. The scheme integrates various network resources, including bandwidth and cache memory, in order to find their unified contribution to minimising delay. The resulted information was then utilised to enhance the routing performance.

In summary, several existing literature approaches focus on taking advantage of the integration of ML and SDN to achieve better network QoS. However, only few prior studies [8,24,27,28] considered the improvement of end-users' QoE via route-based learning. This paper aims to address this issue by introducing a framework that realises improving the satisfaction degree of end-users towards multimedia services by utilising SDN characteristics with RL, which intelligently decides the best path for multimedia flows. The proposed scheme follows the approach presented in [26], in terms of directly learning the route hop-by-hop. It also considers four link-state metrics (i.e., available bandwidth, delay, jitter and packet loss rate) as parameters for the RL agent to deliver video streaming packets. The proposed RL-based solution learns the next route, hop-by-hop, for video packets at each OpenFlow switch from the source to the destination while prioritising links' available bandwidth and avoiding links with high delay, jitter and packet loss rate.

### 3. RL-Based Multimedia Traffic Routing Architecture

In this section, the architecture for video streaming traffic based routing in SDN environments is introduced to enhance the QoS of the network with an emphasis on optimising users' QoE. Figure 1 depicts the architecture of RL-based multimedia traffic routing.

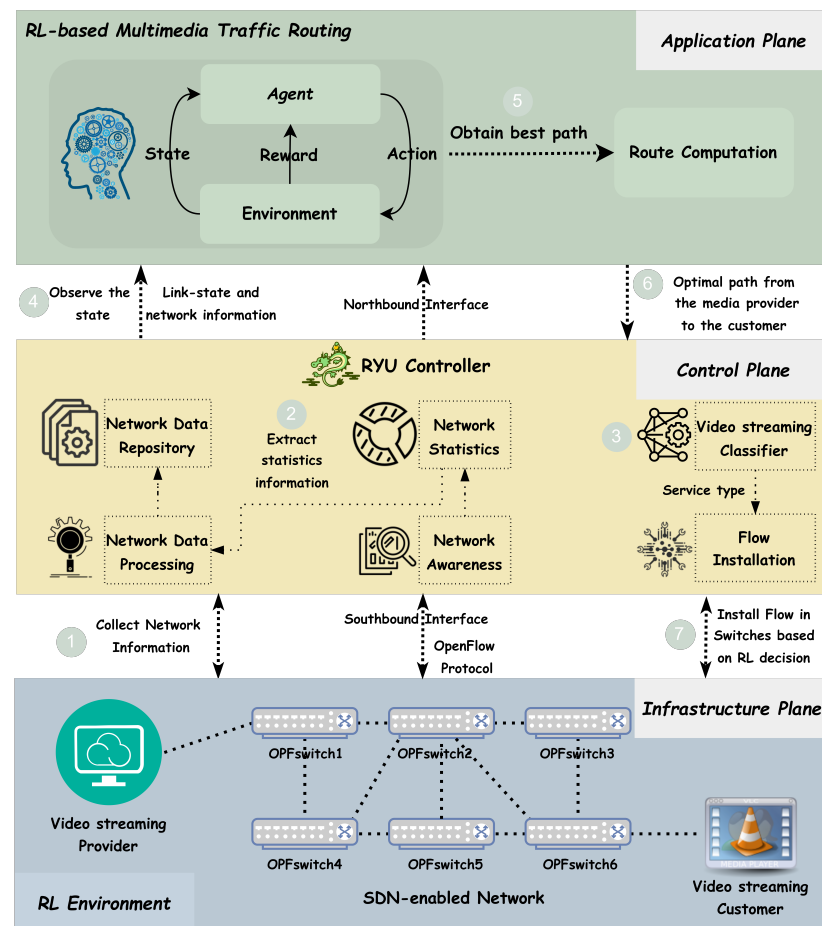


Figure 1. RL-based multimedia traffic routing architecture.

### 3.1. Architecture and Components

#### 3.1.1. Infrastructure Plane

This plane includes three major components ① Video streaming providers ② SDN-enabled network ③ Video streaming customers. The first component is responsible for providing various video streaming services, which will be transferred over the SDN network. The second one contains the forwarding switches, which are under the management of the SDN controller and the links connecting them. These elements intend to carry out a set of primary tasks, such as forward the incoming packets to one or more ports or drop the packets. The video traffic is transmitted based on the path information installed in the flow tables of the SDN switches. The SDN switches have no knowledge about the network and account on the control plane and the application plane to occupy and install their flow tables. The final element is in charge of giving feedback. The feedback indicates the overall level of satisfaction with video streaming services which reflects the QoE of the services. The infrastructure plane also represents a reinforcement learning environment that periodically provides information about network topology by reacting to queries received from the control plane.

#### 3.1.2. Control Plane

This plane connects with the infrastructure plane through the southbound interface. It has a global view of the infrastructure plane since it utilised to gather information about network topology to obtain the environment states. It contains six modules. ① Network awareness; maps the SDN-enabled network topology into a graph representation and stores its physical information to use them in other modules. ② Network statistics; maintains the flow state within the network environment (SDN-enabled network) by periodically gathering statistical information of all flows. ③ Video stream classifier; classifies the network traffic flows where video flows are prioritised against the best-effort traffic generated by using Iperf testing tool [31]. ④ Flow installation; operates re-actively which means that the OpenFlow switches in the infrastructure plane are re-actively programmed to build flow entries once traffic arrives. ⑤ Network data processing; receives network information from the network statistics module and periodically computes network data represented by the network QoS parameters, including link delay, jitter, loss ratio and available bandwidth. ⑥ Network data repository; stores the four QoS metrics processed by the network data processing module. The storage module carries records that outline the source and destination nodes plus the related tuples of QoS metrics.

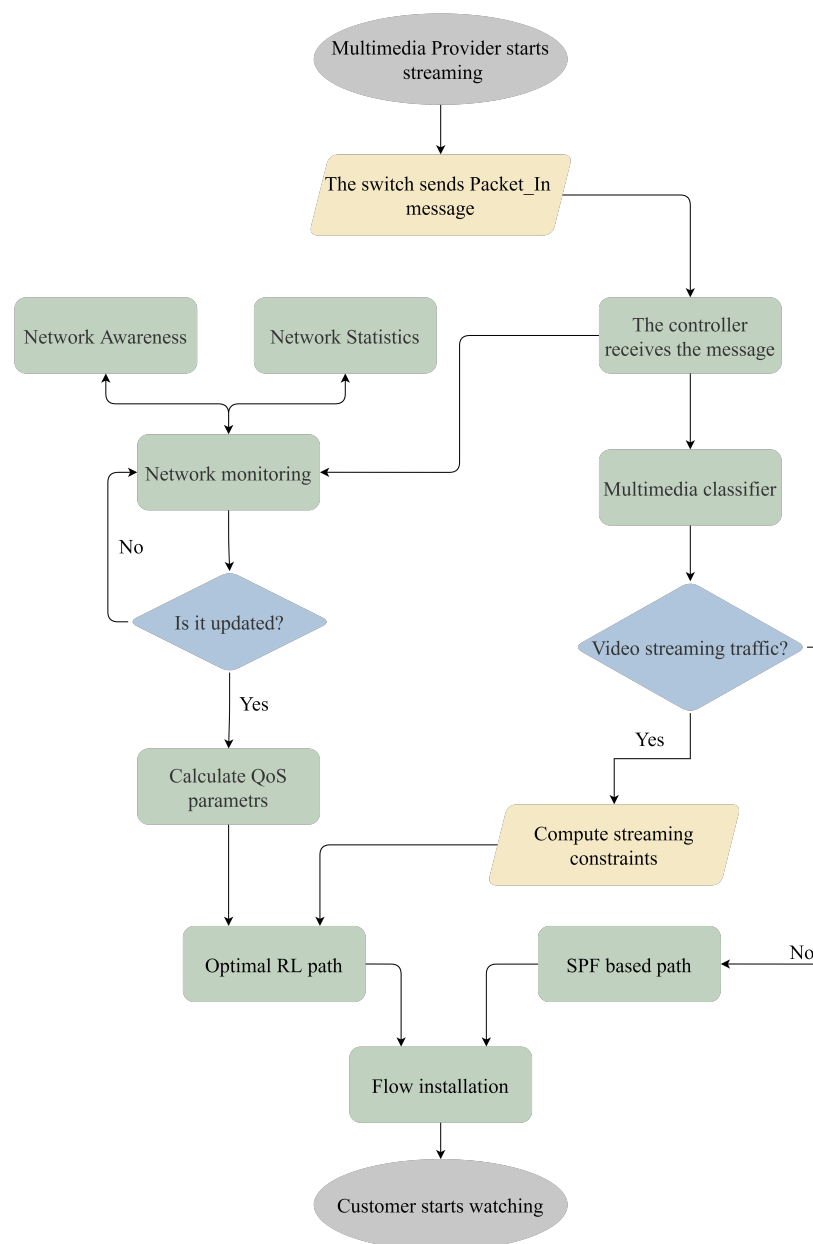
#### 3.1.3. Application Plane

This plane contains the RL method, which learns the network characteristics and applies intelligence concerning route calculation. It communicates with the control plane via northbound interface to get the link-state and topology information in order to compute the optimal route between the source-destination pairs.

### 3.2. Process Description

To begin with, a customer requests to watch a video from a multimedia provider. The SDN switch sends a packet\_In message request to the controller, which starts to monitor the network topology in the infrastructure plane by utilising the network awareness and the network statistics. In the meantime, the controller (on-the-fly) uses the video stream classifier module to check the packet type and decide which routing path should be chosen. There are several scheduled events for monitoring the network. Each event runs periodically based on the utilised topology. Three different network sizes have been used; small, middle and large scale (see Section 5.1). Several experiments under the operation of varying topology sizes have been carried out to select the correct values for each scheduled event which resulted in maintaining a full image of the network state to obtain accurate results. The monitoring step updates every 10 s under the employment of the small-scale network, 15 s when the middle-scale network is utilised, and 20 s when the large-scale

topology is used. It involves the network awareness module, which operates every 5, 6 and 8 s for small, medium and large scale networks, respectively and the network statistics module, which runs within the monitoring step. The monitoring periods include the computation of link delay, measured by implementing a python-based RYU application [32] that operates every 8, 10 and 15 s according to the network size. At this point, it is possible to measure the link-state metrics (i.e., available bandwidth, delay, jitter and packet loss rate), which are later fed as QoS parameters into the RL agent. The measurement of these metrics is explained in detail in Section 4.1. After that, the network data processing and the network data repository are utilised to extract and log the QoS parameters. Based on the classification result, video flows are prioritised against background traffic flows. The application plane receives the topology information and QoS metrics, which allow applying intelligence to find the best route. Once the path is found, it will use the flow installation module, which starts to install the optimal route between the customer and the video provider. Figure 2 illustrates the flowchart of the described process.



**Figure 2.** Flowchart of the proposed RL-based multimedia traffic routing.

## 4. RL-Based Decision Making Solution

### 4.1. Problem Domain

According to graph theory, let the infrastructure plane be modeled into an undirected graph  $G(V, E)$ , where  $V$  represents the set of nodes and  $E$  is the set of links between the nodes.  $V$  is associated with four subsets  $V = (W, X, Y, Z)$ , where  $W$  refers to video streaming customers,  $X$  indicates the video streaming servers,  $Y$  denotes OpenFlow switches, and  $Z$  is the SDN controller. Each customer  $w \in W$  is connected to a forwarding switch  $y \in Y$  through a link  $l \in E$ . Each  $l$  in the network is related to a limited capacity  $C_l$ , allocated to the link flows, and it defines the maximum possible flow allowed to travel via the link. Every traffic flow  $f$  is associated with two types of flows  $F = (F_v \cup F_b)$ , where  $F_v$  stands for video streaming flows and  $F_b$  indicates UDP traffic flows. Generally, a flow in the network is known as a sequence of communications between the source and destination nodes. It is typically defined by its 5-tuple attributes (source IP, destination IP, source port, destination port, protocol field).

The clients request to watch a video from a streaming provider over the SDN-enabled network. The idea is to set up a feasible routing path for the video flows. If  $P$  contains the set of potential routes, then the routing method is employed to obtain the feasible path  $p \in P$ , in which a path is specified by a set of links  $p = l_1, \dots, l_n$  that connects the nodes between source-destination pairs. If more traffic flows exist in the same link and compete over the network link capacity, the link gets heavily loaded and congested since the traffic flows trying to pass through this link exceed its bandwidth. In this case, the involved traffic flows are more likely to exhibit higher data loss and delay, which seriously impact the network QoS and users' QoE. Accordingly, this study presents an optimisation method that addresses this case by considering a network monitor, where network status is periodically monitored, and path decisions will be deployed according to the current status of network links. Therefore, RL agent will react quickly to congestion in the link and suggest another path for video transmission. In this case, the proposed solution shall provide a good viewing experience to the clients.

The proposed solution of video streaming QoS/QoE optimisation considers four link-state metrics in each path  $p$ : link available bandwidth  $B_l$ , link delay  $D_l$ , link loss rate  $L_l$  and link jitter  $J_l$ . These parameters may cause distortions in the video streaming that affect the received service quality [33]. The resolution of the video is affected by the available bandwidth of the network. Having enough bandwidth will ease the transmission of the video smoothly hence providing better video quality. Whereas, the lack of available bandwidth affects customers' QoE as it leads to video streaming quality degradation [34]. To obtain the  $B_l$  through an end-to-end path between two pairs, the network statistics in the control plane is used to track the traffic flow and collect the received and transmitted bytes at each port at regular intervals of time. Compared with the retrieved values at two successive responses time, it is feasible to calculate the available bandwidth. Let say that the controller receives *OFPortStatsReply* message from the infrastructure plane, at time  $t_1$ , which contains the number of bytes received,  $br_{t_1}$ . Then, after an interval period  $\Delta t$ , another *OFPortStatsReply* message is obtained, at time  $t_2$ , which contains the number of bytes received  $br_{t_2}$ , the used bandwidth can be computed as follows:

$$ubw_l = \frac{br_{t_2} - br_{t_1}}{\Delta t} \quad (1)$$

where  $\Delta t$  represents the period of the sample interval. Then, the available bandwidth  $abw_l$  of link  $l$  is decided by:

$$abw_l = C_l - ubw_l \quad (2)$$

where  $ubw_l$  is the total throughput of the passing flow  $f \in F$  in  $l \in E$ .

The delay impacts the video quality, particularly at the beginning of viewing, causing the start-up delay. However, this impact can be minimised in the rest of the video delivery due to the existence of a buffer which helps to play the video smoothly [35]. The measurement of  $D_l$  follows the method published in [36]. RYU controller  $Z$  sends

a Link Layer Discovery Protocol (LLDP) packet to a source switch,  $y_1 \in Y$  and records the sending timestamp.  $y_1$  then sends the LLDP packet to a destination switch  $y_2$  and from  $y_2$  to  $Z$ .  $Z$  then obtains the total delay ( $T_d$ ) by calculating the difference between sending and receiving times of the LLDP packet. At this moment, the  $T_d$  of the path has been registered  $Z - y_1 - y_2 - Z$ . The next step is to compute the delay between  $Z - y_1$  and  $Z - y_2$ .  $Z$  sends an *OFPEchoRequest* contains the sending time  $ST$  as the data.  $y$  responses with *OFPEchoReply* message and records the receiving time  $RT$ . In this case, the delay between controller and switch  $D_{Zy}$  is calculated based on the following formula:

$$D_{Zy} = \frac{ST - RT}{2} \quad (3)$$

Now, the calculation of the delay between  $y_1$  and  $y_2$  is presented as follows:

$$D_{l_{y_1, y_2}} = T_d - D_{Zy_1} - D_{Zy_2} \quad (4)$$

Link loss affects the received video streaming as it reduces its resolution. In this case, a frame may be lost, resulting in freezing in the most recent frame and jumping to the subsequent consecutive frame that arrives, making the video streaming inefficient to watch. Network congestion usually leads to packet loss [35]. To compute  $L_l$ , statistics from the  $y \in Y$ , have queried using *OFPPortStatsRequest* messages. By taking advantage of these statistics, the delta of transmitted packets by  $y_1$  and received packets by  $y_2$  can be computed, which results in the link loss ratio:

$$L_l = \frac{tx_{op} - rx_{ip}}{tx_{op}} \quad (5)$$

where  $tx_{op}$  is defined as the number of transmitted bytes of the output ports that can be obtained when receiving a reply message. And  $rx_{ip}$  is the number of received bytes of ingress ports that can be collected when another reply message is received in a different period.

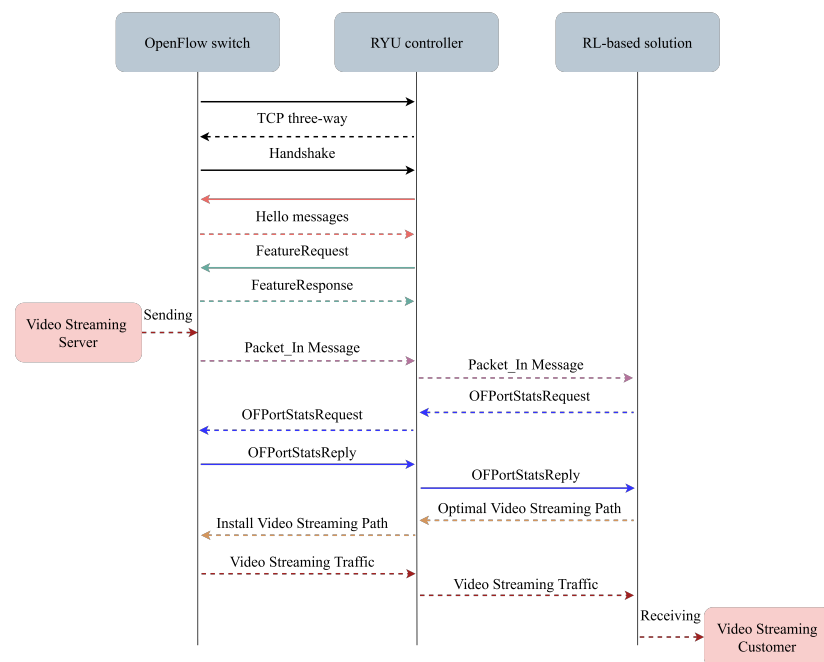
Jitter, also known as delay variation, makes video streaming packets work inefficiently. It affects the presentation of the correct order of the frames by making them wait in the queue. This may cause freezing for the most recent frame until the arrival of the overdue frame, which starts playing in brief to provide time preservation of other arrived frames [37]. The jitter is defined as the difference in the link delay  $D_l$ :

$$J_l = D_{lt_2} - D_{lt_1} \quad (6)$$

where  $t_1$  and  $t_2$  denote respectively the previous and the current link delay.

An explanation of the exchanged messages protocol between the forwarding switch  $y$  with the Ryu controller  $Z$  and the RL-based solution is depicted in Figure 3. It contains the preliminary negotiation and the important messages between  $y$  and  $Z$ . When a multimedia provider begins streaming,  $Z$  receives a packet\_In from  $y$ .  $Z$  utilises its modules to monitor and extract the statistical network information. Then, it passes this request to the application plane where the routing computation takes place. In the end,  $Z$  uses the flow installation module which starts to install an optimal path for video traffic between the two pairs.

With the terms explained earlier, the research focuses on QoS based network optimisation and improving users' experience. The enhancement criteria in this work can be seen as maximising all links utilisation of video streaming based networks. The main strategy to achieve this goal is to avoid paths with high end-to-end delay, jitter and loss ratios while maximising the capacity by prioritising paths with high available bandwidth. Using a Reinforcement Learning (RL) algorithm could be beneficial to realise an intelligent video streaming routing that considers the above problem. RL interacts with the dynamic environment and allows the agent to explore the state collected by the control plane according to an action performed by the agent to acquire the policy that maximises the long term reward.



**Figure 3.** Sequence diagram of OpenFlow with RYU controller and RL-based solution.

#### 4.2. RL-Based Solution

The proposed RL-based intelligent video streaming routing aims to enhance network QoS and user QoE. The RL agent is designed to learn a strategy through interactions with its dynamic environment by repeatedly exploring and observing its state. Based on the available knowledge, it will take actions meant to maximise its total cumulative reward. From this context, the goal of the RL-based solution is to find the optimal path for video streaming-based traffic via interacting with the SDN environment to maximise user QoE concerning the QoS demands of each service. The RL agent employs the Q-learning algorithm to figure out the optimisation problem. Q-learning is a model-free RL algorithm used to find the optimal policy by learning the optimal Q-values per state when all potential state-action pairs are visited for a pre-defined number of loops [38,39]. In the following subsections, the details of the state space, the action space, the exploration-exploitation strategy and the reward function employed to model the proposed problem are described.

##### 4.2.1. State Space

The state in the proposed RL-based solution indicates the nodes of the environment in the infrastructure plane which reflects the OpenFlow switches. The transition from one state to another represents the links connecting the corresponding switches. The network awareness module in the control plane designed to map the SDN-enabled network to the graphical structural representation and stores its physical information to be used by the RL agent. In this case, each state in the state space represents an OpenFlow switch of the network topologies that will be employed for the experimental evaluation of our approach.

##### 4.2.2. Action Space

The action in the proposed RL-based solution determines the selection of a switch neighbour to be the next forwarder to deliver video streaming packets to a destination. It also determines the optimal video streaming traffic policy. The policy is intended to minimise the reward value in the Q-routing method. In the process of assigning a routing path for video streaming flows, the agent learns to choose routes with low delay, jitter and loss ratio while prioritising paths with high available bandwidth. Moreover, the actions should be modified according to the reward value.

In Q-learning, the RL agent visits all states and tries different actions to approximate the optimal Q-function. It then updates and saves the Q-value after an episode in a Q-table which



becomes a reference for the RL agent to select the best route for a node pair. The Q-value is defined as a measure of the overall expected reward if the RL agent is in a state  $S$  and takes action  $A$ . The learning phase of the RL agent includes a series of stages named episodes  $(0, 1, 2, \dots, n, \dots)$ . Throughout the  $n$ th episode at time  $t$ , the agent determines an action  $A$  on a video streaming packet at a current state  $S$  and obtains a reward  $R$  as it proceeds to the next state,  $S'$ . The optimal multimedia traffic routing problem can be modeled as a minimisation problem, and the Q-learning algorithm must be adjusted to adapt to minimisation requirements. The RL agent utilises the following modified Q-learning equation to update the Q-value for optimal path routing.

$$Q(S, A) = \underbrace{Q(S, A)}_{\text{Current Q-value}} + \alpha \underbrace{[R(S, A)]}_{\text{Reward}} + \underbrace{\min_{A'} Q'(S', A')}_{\substack{\text{Minimum predicted} \\ \text{reward, given new} \\ \text{state and all} \\ \text{possible actions}}} - Q(S, A) \quad (7)$$

where  $\alpha$  is the learning rate that decides how much of the new learned value will be utilised. As it can be seen from Equation (7), the output is the new Q-value for the state which is resulted by increment the current Q-value by  $\alpha$  multiplied by the chosen action's Q-value.  $\alpha \in [0, 1]$ , when the value  $\alpha$  is set to 0, the RL agent will not be able to learn from new actions. Contrariwise, if it is set to 1, the agent totally passes over previous knowledge and only values the recent learned information taking into consideration the instant reward for the state-action pair. Higher  $\alpha$  values allow newly learned Q-values to change quickly.

#### 4.2.3. Exploration-Exploitation Strategy

In RL, exploration and exploitation are both very significant concepts. An exploration means a selection of actions other than the ones that have been experienced before while an exploitation indicates the selection of the optimal actions.  $\epsilon$ -greedy strategy was implemented in the proposed solution to have a balance between exploration and exploitation.  $\epsilon$ -greedy method continues to explore, meaning that it gives a chance to execute random actions. It employs a tuning parameter,  $\epsilon \in [0, 1]$ , to indicate whether the agent should explore with a probability of  $\epsilon$  and exploit with a probability of  $1 - \epsilon$ .

The calculation of the  $\epsilon$ -greedy strategy is shown in Equation (8). The RL agent utilises this equation to select the upcoming action at a particular state. In the proposed RL-based solution, the action with the lowest Q-value is selected; this means that, rather than finding a route with the highest reward, the proposed solution obtains a route with the lowest costs by greedily choosing actions having the lowest rewards.

$$A' = \underset{a \in A}{\operatorname{argmin}} Q'(S, A) \quad (8)$$

where  $\underset{a \in A}{\operatorname{argmin}}$  indicates the exploitation of  $Q(S, A)$  in regards to action  $A$ . The ongoing exploration in a greedy way improves the instant reward.

#### 4.2.4. Reward Function

The reward in the proposed RL-based solution is defined in Equation (9), and it is designed to find the best video streaming path according to the four QoS metrics: link available bandwidth, delay, jitter and loss rate. The intention is to find a route with minimum delay, loss, and jitter while prioritising links with large bandwidth to enhance the QoE of the video streaming service. It is noteworthy that different influence factors (IFs) can impact the satisfaction degree of video streaming services. This research concentrates on the network-related IFs, which refer to the QoS factors because it has been demonstrated that these metrics significantly influence the QoE [40].

$$R = w_1 * \frac{1}{B_l} + w_2 * D_l + w_3 * J_l + w_4 * L_l \quad (9)$$

It is worth noting that these metrics have different units (e.g., the available bandwidth in bit per second and delay in millisecond), which have an impact on the learning efficiency thus, each metric value is normalised to united ranges and scales [41]. The values  $w_1, w_2, w_3$ , and  $w_4 \in [0, 1]$  are tuning weights assigned to a determined metric during the reward computation, where  $w_1 + w_2 + w_3 + w_4 = 1$ . Note that the tuning weights of QoS factors are specified in the agreement with the quality standard bounds and their relative importance degree, which are provided in [34] in this fashion,  $L_l$  58.9%,  $J_l$  15.1%,  $D_l$  14.9% and  $B_l$  11.1%.

#### 4.3. RL-Based Multimedia Traffic Routing Algorithm

The proposed routing algorithm is implemented to find the optimal path from the media provider to the customer in the infrastructure plane. Algorithm 1 takes the network topology, the QoS parameters, learning rate  $\alpha$ ,  $\epsilon$ -greedy parameter and the number of learning episodes as input. When video packets flow in the network from a given provider to the desired customer, the algorithm initialises the Q-values of the Q-table to zeros. This means that for a given video packet at the source OpenFlow switch  $y_{src}$ , the first learning episode begins with initialising the state of a video packet at the source switch. At this point, this state starts selecting one action  $A$  from the current state  $S$  using  $\epsilon$ -greedy exploration-exploitation policy. Next, the algorithm uses the network QoS parameters and the state  $S$  to compute the reward based on Equation (9) related to the action  $A$  and discovers the new state  $S'$ . Following that, the Q-function is obtained using Equation (7) and set the next state  $S'$  as the current state  $S$ . The state transition iteration continues as this episode ends, and a new one starts till  $S$  is equivalent to the final state (i.e., the video packet arrives at the OpenFlow destination switch  $y_{dst}$ ). Finally, the RL agent obtains the optimal path that achieves the lowest Q-values derived from the Q-table to forward video packets between the given provider-customer pairs. Once the final goal is reached, the flow installation module in the control plane receives the path and installs it in the routing table of the OpenFlow switches.

---

#### Algorithm 1 Q-Learning-based Multimedia Traffic Routing

---

**Input:** Network topology:  $G(V, E)$   
 Network QoS parameters:  $B_l, D_l, J_l, L_l$   
 Learning rate:  $\alpha$   
 Exploration-exploitation parameter:  $\epsilon$   
 Learning episodes number:  $n$

**Output:** Optimal path from the media provider to the customer that produced the lowest Q-values derived from the Q-table

```

foreach Video transmission over  $(y_{src}, y_{dst}) \in Y$  do
  Initialise Q:  $Q(S, A) = 0, \forall s \in S, \forall a \in A$ 
  for episode  $\leftarrow 1$  to  $n$  do
    Initialise state  $S = y_{src} \in S$ ;
    while  $S'$  is not  $y_{dst}$  do
      Choose an action  $A$  from  $S$  using  $\epsilon$ -greedy policy, Equation (8) derived from Q;
       $R' \leftarrow R(S, A)$  // The RL agent obtains the reward from network QoS parameters using Equation (9); then it observes the new state  $S'$ ;

       $Q(S, A) = Q(S, A) + \alpha[R(S, A) + \min_A Q'(S', A') - Q(S, A)]$ 
       $S \leftarrow S'$  // proceed with the next state;
    end
  end
end
  
```

---

## 5. Evaluation

In this section, an experimental platform used to evaluate the performance of the RL-based solution approach is presented. The proposed architecture implemented on Ubuntu 16.04 installed in HP Z230 tower workstation with an Intel Xeon processor and 16 GB RAM. Mininet emulator [42] used to run the infrastructure plane, which includes the SDN-enabled video streaming network. RYU Controller [32] utilised to emulate the control plane, which collects information about network topology to obtain the environment states.

### 5.1. Test-Bed Preparation

Three realistic network topologies are used for the experimental evaluation of our approach; a modified Abilene topology utilised in [43], Geant [44], and Cernet [45]. The topologies have been built and implemented in Mininet using a Python script; SDN-Openflow switches replaced the nodes for each network topology. Each switch has a host that forwarded and received different types of traffic. Multimedia providers are deployed in a number of Openflow switches. The provider is able to stream real-time Dynamic Adaptive Streaming over HTTP (DASH) based video flows. Due to Mininet resources constraints, links capacities for each topology have been scaled to meet the experimental environment requirements. Table 1 shows the utilised network topologies employed to evaluate our RL-based multimedia traffic routing.

**Table 1.** Three realistic network topologies.

Name	Nodes	Links
Cernet (large-scale topology)	36	48
Geant (middle-scale topology)	23	37
Abilene (small-scale topology)	12	20

### 5.2. QoE Metrics Measurements

The full reference (FR) model is an objective quality assessment utilised to estimate the QoE of video streaming. It performs a direct comparison between the video under processing called distorted and the actual video, named reference, in order to evaluate the video streaming quality. The two assessed videos are studied according to their properties frame-by-frame to inspect different characteristics, including colour processing and contrast features [46]. The Video Multimethod Assessment Fusion (VMAF), presented by Netflix [47] and the Structural Similarity Index Metric (SSIM) [48] are employed as FR measurement metrics for QoE assessment. Both metrics correlate well with human perception and allow an efficient computation. The metrics are computed off-line and as the average VMAF and SSIM over all the video frames. Table 2 represents a mapping of objective QoE (VMAF) and (SSIM) to the nominal Mean Opinion Score (MOS). MOS is a 5-point scale utilised to evaluate the end-users' satisfaction in a subjective manner [21].

**Table 2.** VMAF and VMAF to MOS mapping.

MOS	VMAF	SSIM
5 (Excellent)	80–100	>0.99
4 (Good)	60–79	≥0.95 & <0.99
3 (Fair)	40–59	≥0.88 & <0.95
2 (Poor)	20–39	≥0.5 & <0.88
1 (Bad)	<20	<0.5

### 5.3. Learning Parameters Settings

Before implementing the proposed RL-based solution, it is essential to determine the learning rate  $\alpha$  and exploration probability  $\epsilon$  values. With a probability of exploration value close to one and a high learning rate value, the RL agent managed to find shorter

routes to deliver the video streaming traffic along the sequence of episodes. After several experiments, the parameters are set as follows:  $\alpha = 0.9$ ,  $\epsilon = 0.8$ , and the number of training episodes is set to 300.

#### 5.4. Evaluation Scenarios

Several testing scenarios are implemented to analyse the proposed RL-based solution algorithm. They demonstrate the importance of QoS parameters and their impact on video streaming quality perceived by customers. The goal is to evaluate the well-known routing algorithms such as Shortest Path First (SPF) with the proposed approach under varying traffic load with different network topologies scenarios. SPF is a primary mechanism to originate routing paths, which is extensively utilised in multiple protocols like Open Shortest Path First (OSPF). OSPF-based approach is compared with our solution to evaluate customers desired satisfaction across real-time DASH video streaming using both approaches. In each test, the reference video without degradation is distorted and recorded to generate the processed video with degradation. At this stage, by having both videos, an objective experiment of the perceived video quality is conducted. As an outcome of this experiment, VMAF and SSIM values are obtained.

### 6. Results and Discussions

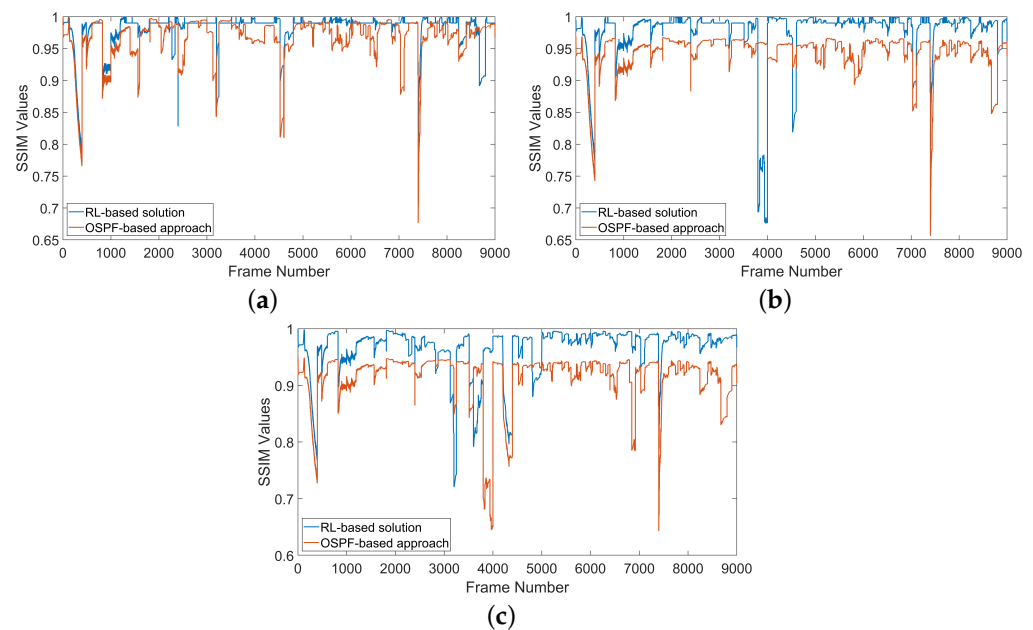
This section explores the impact of our proposed solution on the users' perceived quality when both low and high traffic loads existed. Three different topology scales are used as depicted in Table 1. Iperf testing tool was also used to inject high and low traffic loads into the simulated network. Real-time DASH-based video flows is utilised to test the performance of the RL-based solution. DASH video is divided into 4-s chunks encoded into five discrete bit rates ranging from 260 Kbps to 2998 Kbps using FFmpeg version 4.3.2 with the H.264 codec, and segmented based on GPAC MP4Box in order to create the DASH manifest and associated files. The video content streamed by multimedia providers is the "Big Buck Bunny" animation [49] with a  $1920 \times 1080$  pixels resolution and was cut into 5 min long. The selection of hosts that partake in the experiment has been constructed to enable the traffic flow to pass through the whole network topology. In the meantime, the study has utilised Wireshark as video traffic monitoring software in the end-user's device in order to capture the received video segments during video streaming. Once the streaming ended, the monitoring step on the client device is terminated and saved on the PCAP file. It is noteworthy that the video that was finally broadcast is not available; therefore, the PCAP file is used to collect video segments and recreate the processed video. Now that the processed video is ready, the objective study to determine the QoE metrics values is possible to be executed.

#### 6.1. The Impact of Low Traffic Load on Client Satisfaction

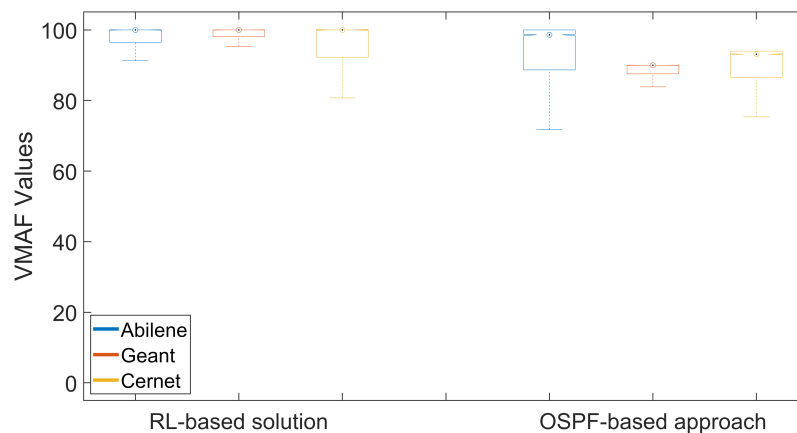
This part shows the impact of low traffic load on the end-users' perceived QoE under three different topology scales. It presents the performance comparison of the proposed RL-based solution with the OSPF protocol in terms of the customers' satisfaction represented by SSIM and VMAF, network throughput and packet loss rate and delay variation for the generated traffic.

SSIM and VMAF correlate well with the customer perception and allow an efficient calculation reflecting the clients' QoE. It can be seen from Figure 4 the obtained SSIM values for DASH video with the proposed RL-based solution maintained a high score in all the three topologies, representing almost an excellent viewing behaviour. The average SSIM values produced with the proposed scheme reaches 98% under the three networks. In the case of the OSPF protocol, QoE was reduced to the range between good and fair for the middle and large scale networks. However, with Abilene, the OSPF-based approach tends to have a slightly similar viewing experience to the RL-based solution. In addition, Figure 5 illustrates the average VMAF scores for DASH video with both schemes. It can be observed that RL-based solution improved the user-perceived quality, specifically under Geant and

Cernet networks, and presents high scores (an average of 93 under all networks) leading to an excellent viewing experience.



**Figure 4.** SSIM values for DASH video under low traffic loads in three different topologies. (a) Abilene; (b) Geant; (c) Cernet.



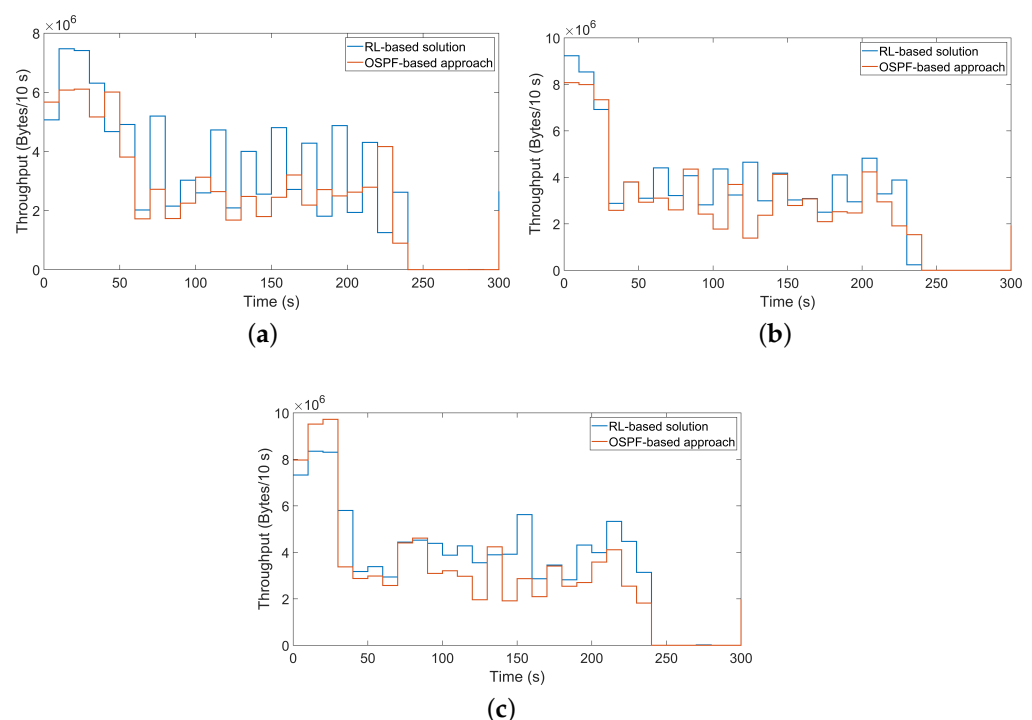
**Figure 5.** VMAF values for DASH video under low traffic loads in three different topologies.

Table 3 shows that the RL-based solution produced a lower packet loss rate for the background traffic compared to the OSPF-based approach. With respect to the OSPF protocol, UDP traffic experiences higher data loss as the topology changes the scale from a small to a large network. For instance, the generated traffic between H7–H35 introduced a 0.10% loss rate in the OSPF-based approach since 128 packets were dropped out of 127,551. In comparison, the RL-based solution decreased the loss rate as only 28 packets were dropped out of 127,551, leading to a 0.022% loss rate. It can be observed that the improvement of packet loss rate with the RL-based solution occurs in all of the client-server applied in the experiment under various topology scales. However, The RL-based solution may impose delay since complexity lies in the dynamic selection of the optimal path for video transmission introduced by the RL agent, which forces the UDP traffic to take another route. In this experiment, the average packet jitter between client and server is monitored with our solution compared to the OSPF-based approach. As expected, it can be clearly noticed that the OSPF-based scheme presented less jitter in almost all of the client-server routes under various network topologies.

**Table 3.** The number of packets dropped and average jitter during video transmission for the low background traffic load under three topologies (includes random client-server from each network topology).

Network Topology		Abilene		Geant		Cernet	
Client-Server		H5-H1	H10-H4	H15-H6	H18-H9	H7-H35	H22-H19
OSPF-based approach	Packets dropped (out of 127,551)	39	73	84	117	128	57
	Average packet jitter (in ms)	0.008	0.017	0.013	0.015	0.022	0.004
RL-based solution	Packets dropped (out of 127,551)	45	41	29	1	28	51
	Average packet jitter (in ms)	0.010	0.015	0.018	0.017	0.008	0.019

Figure 6 demonstrates the throughput of DASH video streaming when the network encounters low traffic loads. As noticed, the throughput drops with the OSPF-based scheme due to the increase in the packet loss rate. Although, in all the three topologies, both approaches present almost a similar performance, between 0–50 s. However, as the video streaming transmission continues, with the low background traffic running on the network, the throughput drops and remains stable at 2.8 Mbps for the OSPF-based approach. At this point, the customer faces re-buffering events because of throughput decrease situations that reduce the client QoE. Contrariwise, the results reveal a higher throughput with the proposed RL-based solution and the stability achieved at more than 3.5 Mbps in all the three topologies. The RL agent repeatedly interacts with the SDN *Infrastructure Plane*, avoids paths with a high data loss rate, and prioritises paths with large available bandwidth, enhancing network throughput and providing a better video quality.



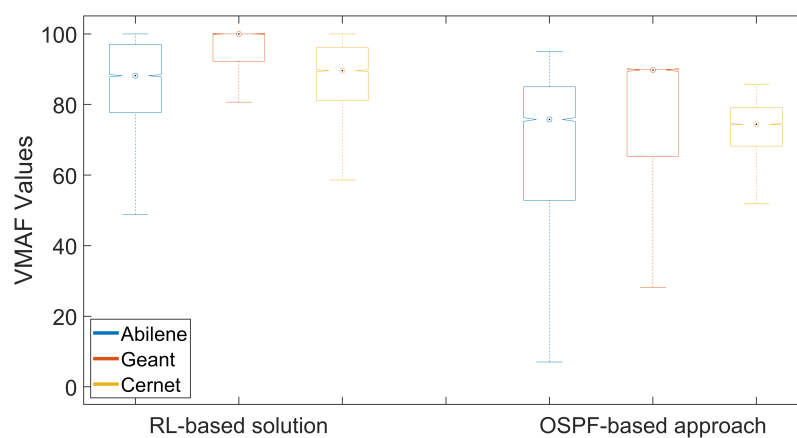
**Figure 6.** Network throughput during video transmission under low traffic loads in three different topologies. (a) Abilene; (b) Geant; (c) Cernet.

## 6.2. The Impact of High Traffic Load on Client Satisfaction

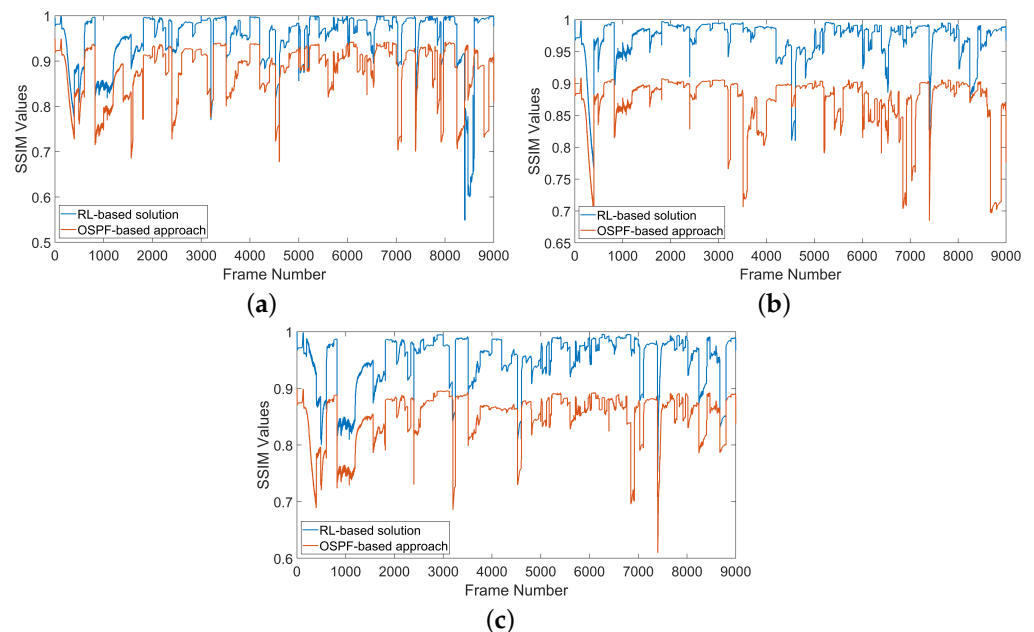
This part illustrates the impact of high traffic load on the end-users' perceived QoE under three different topology scales. The same performance comparison and the same video sample utilised under low traffic were applied; however, the bandwidth of the background traffic that is introduced in the three networks using the Iperf tool is increased.



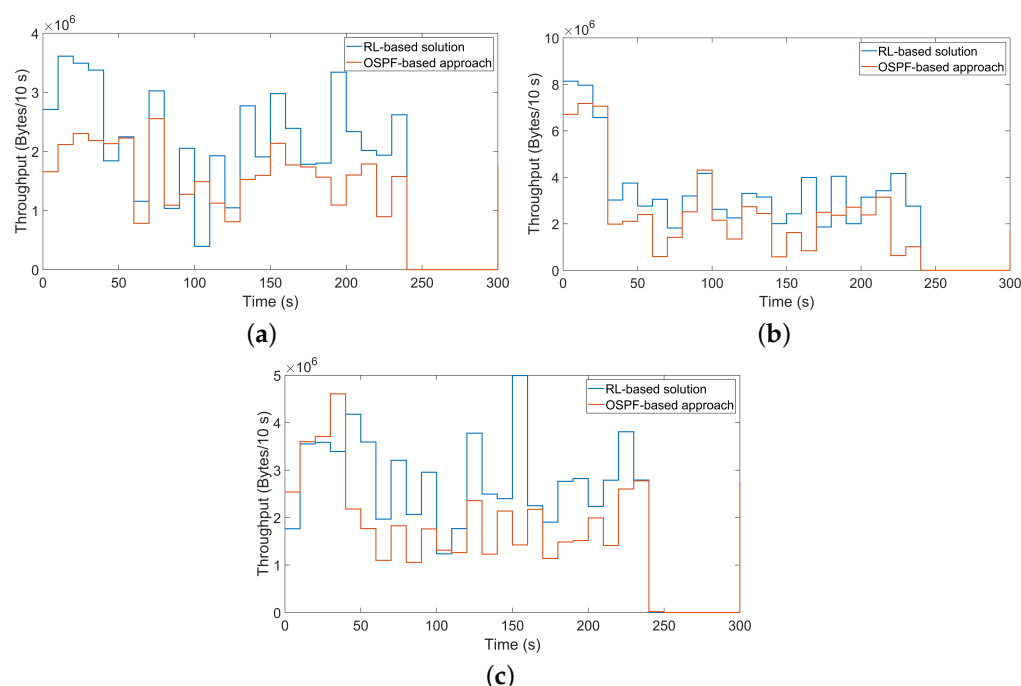
As the high traffic load operates, the network gets overloaded, which causes traffic flows to exhibit data losses. Nevertheless, SSIM and VMAF results indicate that our proposed solution performs better than the OSPF-based approach despite the user-perceived quality dropping to good under Cernet and Abilene topologies. The results with the middle-scale network shows an excellent user perceived QoE (see Figure 7). In this case, the viewing experience is affected when the network topology increases. DASH video streaming without the RL-based solution reveals a massive decrease in the users' perceived final video quality, and according to the VMAF and SSIM results, Geant and Cerent users display bad quality viewing behaviour as depicted in Figures 7 and 8, respectively. However, it can be seen in Figure 9a that under small-scale network, the end-user perceived fair viewing experience. As observed, the client satisfaction represented by QoE is dropped with both schemes under large-scale network. However, the results suggest that the obtained QoE of end-users increases with the proposed RL-based solution.



**Figure 7.** VMAF values for DASH video under high traffic loads in three different topologies.



**Figure 8.** SSIM values for DASH video under high traffic loads in three different topologies. (a) Abilene; (b) Geant; (c) Cernet.



**Figure 9.** Network throughput during video transmission under high traffic loads in three different topologies. (a) Abilene; (b) Geant; (c) Cernet.

The results in Table 4 show that the RL-based solution produced a low data loss rate for the generated traffic compared to the OSPF-based approach. It is noticeable that the proposed solution draws advantages when applied on small and middle-scale networks; for instance, the generated traffic under the two networks indicated that the OSPF-based approach reported a high loss rate. Hence, the OSPF protocol cannot resolve the network congestion by rerouting the traffic flows. In contrast, the proposed solution showed a dramatic decrease in the loss rate in these networks. Although the RL-based solution offers better performance in terms of data loss rate under Abilene and Geant, however, considering the results in Table 4 OSPF-based approach imposes lower jitter for almost all the traffic between client-server of both networks. In a large-scale network, the results indicate an improvement in both packet loss rate and delay variation.

**Table 4.** The number of packets dropped and average jitter during video transmission for the high background traffic load under three topologies (includes random client-server from each network topology).

Network Topology		Abilene		Geant		Cernet	
Client-Server		H5–H1	H10–H4	H15–H6	H18–H9	H7–H35	H22–H19
OSPF-based approach	Packets dropped (out of 178,572)	1176	193	1264	1809	196	641
	Average packet jitter (in ms)	0.009	0.010	0.017	0.011	0.017	0.011
RL-based solution	Packets dropped (out of 178,572)	826	123	100	122	146	297
	Average packet jitter (in ms)	0.022	0.026	0.025	0.007	0.008	0.008

The network throughput under high traffic loads is illustrated in Figure 9. As observed, the video streaming throughput decreases with the OSPF-based approach because the network gets congested and the links are highly experiencing data loss; the throughput value hits a massive drop and reaches even below 1 Mbps under all networks. With the proposed RL-based solution, on the other hand, the throughput value significantly increases and hits 4 Mbps under Cernet topology, even if the network size increases, as depicted in Figure 9c; therefore, the client maintains a better viewing experience.

## 7. Conclusions

In this work, a new reinforcement learning-based routing framework for multimedia traffic over SDN has been proposed. The presented approach aims to provide an acceptable QoE for customers in transmitting video on an SDN network. By enhancing QoE, the customer can see the precise and accurate video frames transmitted from the video streaming provider side. The proposed method leverages the capabilities of SDN paradigm to monitor and gather network statistics to calculate the optimal path. Moreover, the RL agent learns to choose a path with minimum packet loss ratio, end-to-end delay, and jitter and prioritises bandwidth to enhance end-user QoE. Based on the experiment results, the RL-based solution outperforms the OSPF-based approach and produces an excellent and good customers' perceived QoE when low and high traffic loads are introduced under various realistic network topologies (i.e., Abilene, Geant, and Cernet). The obtained QoE of DASH video streaming significantly degraded without the proposed solution and the MOS indicates bad quality when high traffic loads existed. Furthermore, the RL-based scheme increased network throughput and decreased packet loss rate under both traffic loads, which resulted in an excellent user-perceived QoE for middle-scale topology under high traffic loads. To further improve the proposed solution scheme, future work needs to consider integrating QoE metrics with network QoS requirement parameters and utilising Deep Reinforcement Learning for routing decisions' enhancement.

**Author Contributions:** Methodology, M.A.J.; software, M.A.J. and A.A.-S.; validation, M.A.J., T.K. and S.T.; investigation, M.A.J., T.K. and W.S.B.; writing—original draft preparation, M.A.J.; writing—review and editing, T.K., S.T., A.A.-S. and W.S.B.; supervision, T.K., S.T., A.A.-S. and W.S.B.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Ministry of Higher Education and Scientific Research, Republic of Iraq, in 6 September 2017 to sponsor Mohammed Al Jameel to pursue his PhD research.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The resulted data underpinning this study are openly available from the University of Northampton at <http://doi.org/10.24339/fbd41ebe-3ba0-4f1a-bd07-3668591e8b77> (accessed on 7 June 2022).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sandvine. *The Global Internet Phenomena Report COVID-19 Spotlight*; Sandvine: Waterloo, ON, Canada, 2020.
2. Trestian, R.; Comsa, I.S.; Tuysuz, M.F. Seamless multimedia delivery within a heterogeneous wireless networks environment: Are we there yet? *IEEE Commun. Surv. Tutor.* **2018**, *20*, 945–977. [\[CrossRef\]](#)
3. Doumanoglou, A.; Zioulis, N.; Griffin, D.; Serrano, J.; Phan, T.K.; Jiménez, D.; Zarpalas, D.; Alvarez, F.; Rio, M.; Daras, P. A system architecture for live immersive 3D-media transcoding over 5G networks. In Proceedings of the 2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Valencia, Spain, 6–8 June 2018; pp. 11–15.
4. Jawad, N.; Salih, M.; Ali, K.; Meunier, B.; Zhang, Y.; Zhang, X.; Zetik, R.; Zarakovitis, C.; Koumaras, H.; Kourtis, M.A.; et al. Smart television services using NFV/SDN network management. *IEEE Trans. Broadcast.* **2019**, *65*, 404–413. [\[CrossRef\]](#)
5. Barakabitze, A.A. QoE-Centric Control and Management of Multimedia Services in Software Defined and Virtualized Networks. Ph.D. Thesis, University of Plymouth, Plymouth, UK, 2020.
6. Martin, A.; Egaña, J.; Flórez, J.; Montalban, J.; Olaizola, I.G.; Quartulli, M.; Viola, R.; Zorrilla, M. Network resource allocation system for QoE-aware delivery of media services in 5G networks. *IEEE Trans. Broadcast.* **2018**, *64*, 561–574. [\[CrossRef\]](#)
7. Comsa, I.S.; Muntean, G.M.; Trestian, R. An innovative machine-learning-based scheduling solution for improving live UHD video streaming quality in highly dynamic network environments. *IEEE Trans. Broadcast.* **2020**, *67*, 212–224. [\[CrossRef\]](#)
8. Huang, X.; Yuan, T.; Qiao, G.; Ren, Y. Deep reinforcement learning for multimedia traffic control in software defined networking. *IEEE Netw.* **2018**, *32*, 35–41. [\[CrossRef\]](#)
9. Grigoriou, E. Quality of Experience Monitoring and Management Strategies for Future Smart Networks. 2020. Available online: <https://iris.unica.it/handle/11584/284401> (accessed on 16 February 2022).
10. Ullah, Z.; Al-Turjman, F.; Mostarda, L.; Gagliardi, R. Applications of artificial intelligence and machine learning in smart cities. *Comput. Commun.* **2020**, *154*, 313–323. [\[CrossRef\]](#)

11. Lekharu, A.; Moulii, K.; Sur, A.; Sarkar, A. Deep learning based prediction model for adaptive video streaming. In Proceedings of the 2020 International Conference on COMMunication Systems & NETworkS (COMSNETS), Bangalore, India, 7–11 January 2020; pp. 152–159.
12. Anand, D.; Togou, M.A.; Muntean, G.M. A Machine Learning Solution for Automatic Network Selection to Enhance Quality of Service for Video Delivery. In Proceedings of the 2021 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Chengdu, China, 4–6 August 2021; pp. 1–5.
13. Kattadige, C.; Raman, A.; Thilakarathna, K.; Lutu, A.; Perino, D. 360NorVic: 360-degree video classification from mobile encrypted video traffic. In Proceedings of the 31st ACM Workshop on Network and Operating Systems Support for Digital Audio and Video, Istanbul, Turkey, 28 September–1 October 2021; pp. 58–65.
14. Anerousis, N.; Chemouil, P.; Lazar, A.A.; Mihai, N.; Weinstein, S.B. The Origin and Evolution of Open Programmable Networks and SDN. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1956–1971. [\[CrossRef\]](#)
15. Egilmez, H.E.; Civanlar, S.; Tekalp, A.M. An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks. *IEEE Trans. Multimed.* **2012**, *15*, 710–715. [\[CrossRef\]](#)
16. Juttner, A.; Szviatovski, B.; Mécs, I.; Rajkó, Z. Lagrange relaxation based method for the QoS routing problem. In Proceedings of the Conference on Computer Communications—Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No. 01CH37213), Anchorage, AK, USA, 22–26 April 2001; Volume 2, pp. 859–868.
17. Yu, T.F.; Wang, K.; Hsu, Y.H. Adaptive routing for video streaming with QoS support over SDN networks. In Proceedings of the 2015 International Conference on Information Networking (ICOIN), Siem Reap, Cambodia, 12–14 January 2015; pp. 318–323.
18. Ongaro, F.; Cerqueira, E.; Foschini, L.; Corradi, A.; Gerla, M. Enhancing the quality level support for real-time multimedia applications in software-defined networks. In Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC), Garden Grove, CA, USA, 16–19 February 2015; pp. 505–509.
19. Rego, A.; Sendra, S.; Jimenez, J.M.; Lloret, J. OSPF routing protocol performance in Software Defined Networks. In Proceedings of the 2017 Fourth International Conference on Software Defined Systems (SDS), Valencia, Spain, 8–11 May 2017; pp. 131–136.
20. Rego, A.; Sendra, S.; Jimenez, J.M.; Lloret, J. Dynamic metric OSPF-based routing protocol for software defined networks. *Clust. Comput.* **2019**, *22*, 705–720. [\[CrossRef\]](#)
21. Elbasheer, M.O.; Aldegheshem, A.; Lloret, J.; Alrajeh, N. A QoS-Based routing algorithm over software defined networks. *J. Netw. Comput. Appl.* **2021**, *194*, 103215. [\[CrossRef\]](#)
22. Uzakgider, T.; Cetinkaya, C.; Sayit, M. Learning-based approach for layered adaptive video streaming over SDN. *Comput. Netw.* **2015**, *92*, 357–368. [\[CrossRef\]](#)
23. Sendra, S.; Rego, A.; Lloret, J.; Jimenez, J.M.; Romero, O. Including artificial intelligence in a routing protocol using software defined networks. In Proceedings of the 2017 IEEE International Conference on Communications Workshops (ICC Workshops), Paris, France, 21–23 May 2017; pp. 670–674.
24. Al-Jawad, A.; Shah, P.; Gemikonakli, O.; Trestian, R. LearnQoS: A learning approach for optimizing QoS over multimedia-based SDNs. In Proceedings of the 2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Valencia, Spain, 6–8 June 2018; pp. 1–6.
25. Hossain, M.B.; Wei, J. Reinforcement learning-driven QoS-aware intelligent routing for software-defined networks. In Proceedings of the 2019 IEEE global conference on signal and information processing (GlobalSIP), Ottawa, ON, Canada, 11–14 November 2019; pp. 1–5.
26. Godfrey, D.; Kim, B.S.; Miao, H.; Shah, B.; Hayat, B.; Khan, I.; Sung, T.E.; Kim, K.I. Q-learning based routing protocol for congestion avoidance. *Comput. Mater. Contin.* **2021**, *68*, 3671. [\[CrossRef\]](#)
27. Al-Jawad, A.; Comşa, I.S.; Shah, P.; Gemikonakli, O.; Trestian, R. An innovative reinforcement learning-based framework for quality of service provisioning over multimedia-based sdn environments. *IEEE Trans. Broadcast.* **2021**, *67*, 851–867. [\[CrossRef\]](#)
28. Al-Jawad, A.; Comşa, I.-S.; Shah, P.; Gemikonakli, O.; Trestian, R. REDO: A reinforcement learning-based dynamic routing algorithm selection method for SDN. In Proceedings of the 2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Online, 9–11 November 2021; pp. 54–59.
29. Guo, Y.; Wang, W.; Zhang, H.; Guo, W.; Wang, Z.; Tian, Y.; Yin, X.; Wu, J. Traffic Engineering in hybrid Software Defined Network via Reinforcement Learning. *J. Netw. Comput. Appl.* **2021**, *189*, 103116. [\[CrossRef\]](#)
30. Liu, W.x.; Cai, J.; Chen, Q.C.; Wang, Y. DRL-R: Deep reinforcement learning approach for intelligent routing in software-defined data-center networks. *J. Netw. Comput. Appl.* **2021**, *177*, 102865. [\[CrossRef\]](#)
31. Gueant, V. iPerf—The Ultimate Speed Test Tool for TCP, UDP and SCTPTEST the Limits of Your Network + Internet Neutrality Test. Available online: <https://iperf.fr/> (accessed on 10 December 2021).
32. Asadollahi, S.; Goswami, B.; Sameer, M. Ryu controller's scalability experiment on software defined networks. In Proceedings of the 2018 IEEE international conference on current trends in advanced computing (ICCTAC), Bangalore, India, 1–2 February 2018; pp. 1–5.
33. Vega, M.T.; Perra, C.; Liotta, A. Resilience of video streaming services to network impairments. *IEEE Trans. Broadcast.* **2018**, *64*, 220–234. [\[CrossRef\]](#)
34. Kim, H.J.; Yun, D.G.; Kim, H.S.; Cho, K.S.; Choi, S.G. QoE assessment model for video streaming service using QoS parameters in wired-wireless network. In Proceedings of the 2012 14th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 19–22 February 2012; pp. 459–464.

35. Chen, Y.; Wu, K.; Zhang, Q. From QoS to QoE: A tutorial on video quality assessment. *IEEE Commun. Surv. Tutor.* **2014**, *17*, 1126–1165. [[CrossRef](#)]
36. Oginni, O.; Bull, P.; Wang, Y. Constraint-aware software-defined network for routing real-time multimedia. *ACM SIGBED Rev.* **2018**, *15*, 37–42. [[CrossRef](#)]
37. Benmir, A.; Korichi, A.; Bourouis, A.; Alreshoodi, M.; Al-Jobouri, L. GeoQoE-Vanet: QoE-aware geographic routing protocol for video streaming over vehicular ad-hoc networks. *Computers* **2020**, *9*, 45. [[CrossRef](#)]
38. Mammeri, Z. Reinforcement learning based routing in networks: Review and classification of approaches. *IEEE Access* **2019**, *7*, 55916–55950. [[CrossRef](#)]
39. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 2018.
40. Juluri, P.; Tamarapalli, V.; Medhi, D. Measurement of quality of experience of video-on-demand services: A survey. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 401–418. [[CrossRef](#)]
41. Al Shalabi, L.; Shaaban, Z. Normalization as a preprocessing engine for data mining and the approach of preference matrix. In Proceedings of the 2006 International Conference on Dependability of Computer Systems, Szklarska Poręba, Poland, 25–27 May 2006; pp. 207–214.
42. de Oliveira, R.L.S.; Schweitzer, C.M.; Shinoda, A.A.; Prete, L.R. Using Mininet for emulation and prototyping Software-Defined Networks. In Proceedings of the 2014 IEEE Colombian Conference on Communications and Computing (COLCOM), Bogota, Colombia, 4–6 June 2014; pp. 1–6. [[CrossRef](#)]
43. Henni, D.E.; Ghomari, A.; Hadjadj-Aoul, Y. A consistent QoS routing strategy for video streaming services in SDN networks. *Int. J. Commun. Syst.* **2020**, *33*, e4177. [[CrossRef](#)]
44. Kirstein, P.T. European International Academic Networking: A 20 Year Perspective. In Proceedings of the TERENA Networking Conference, Rhodes, Greece, 7–10 June 2004.
45. Liu, Y. Current situation and prospect of CERNET. In *China's e-Science Blue Book 2020*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 327–334.
46. Lahoulou, A.; Larabi, M.C.; Beghdadi, A.; Viennet, E.; Bouridane, A. Knowledge-based taxonomic scheme for full-reference objective image quality measurement models. *J. Imaging Sci. Technol.* **2016**, *60*, 60406-1. [[CrossRef](#)]
47. Li, Z.; Bampis, C.; Novak, J.; Aaron, A.; Swanson, K.; Moorthy, A.; Cock, J. Netflix Technology Blog—VMAF: The Journey Continues. 2018. Available online: <http://mcl.usc.edu/wp-content/uploads/2018/10/2018-10-25-Netflix-Worked-with-Professor-Kuo-on-Video-Quality-Metric-VMAF.pdf> (accessed on 25 March 2022).
48. Sara, U.; Akter, M.; Uddin, M.S. Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study. *J. Comput. Commun.* **2019**, *7*, 8–18. [[CrossRef](#)]
49. Big Buck Bunny. Available online: <https://peach.blender.org/> (accessed on 18 January 2022).