

Document downloaded from:

<http://hdl.handle.net/10251/189734>

This paper must be cited as:

Elbasheer, MO.; Aldegheishem, A.; Lloret, J.; Alrajeh, N. (2021). A QoS-Based routing algorithm over software defined networks. *Journal of Network and Computer Applications*. 194:1-14. <https://doi.org/10.1016/j.jnca.2021.103215>



The final publication is available at

<https://doi.org/10.1016/j.jnca.2021.103215>

Copyright Elsevier

Additional Information

A QoS-Based Routing Algorithm over Software Defined Networks

Majda Omer Elbasheer Ali¹, Abdulaziz Aldegheishem², Jaime Lloret^{3,4}, Nabil Alrajeh⁵

¹Sudan University of Science and Technology, Collage of Computer Science and Information Technology, Khartoum, Sudan.

²Urban Planning Department, College of Architecture and Planning, King Saud University, P. O. Box 57448, Riyadh 11574, Saudi Arabia.

³Instituto de Investigación para la gestión Integrada de Zonas Costeras, Universitat Politècnica de Valencia, Camino Vera s/n, 46022, Valencia, Spain

⁴School of Computing and Digital Technologies, Staffordshire University, Stoke, UK

⁵Biomedical Technology Department, College of Applied Medical Sciences, King Saud University, P. O. Box 10219, Riyadh 1433, Saudi Arabia.

¹majda.omer@gmail.com, ²aldeghei@ksu.edu.sa, ³jlloret@dc.com.upv.es, ⁴nabil@ksu.edu.sa

Abstract— Video streaming requires reliable transmission and a steady network to satisfy the video users and applications QoS needs. The network service must provide and meet a certain level of performance to accommodating these QoS, specially in Smart City environments, but current Internet best-effort service does not guarantee QoS. Software-defined networking (SDN) is an approach intended to separate the network control and data (forwarding) planes to get better management and optimization, which clearly enhance the planning and development of smart urban communities. In this paper, we propose a dynamic QoS algorithm over SDN to select the optimum path that meets video QoS in order to optimize the quality of experience (QoE). We differentiate the traffic based on video resolution QoE parameters to Standard Definition (SD) and High Definition (HD). The results demonstrate that the proposed method obtained better viewing quality for an smart community environment and increases the overall network throughput.

Keywords— QoS-Based Routing, software-defined Networks (SDN), video streaming, QoE, Smart Cities, Smart Community.

I. INTRODUCTION

Recently, there has been continuous growth in video consumption by Internet users, accompanied by the development in the technology of real-time video sessions, like IPTV and Internet video services. According to Seufert et al. [1], the video traffic represented 57% of all Internet traffic in 2012, and probably this percentage has doubled these days due to the spread of social media and the diversity of video applications. Undoubtedly, video streaming requiring guaranteed performance which entails a particular requirement from the networks, those types of applications characterized as real-time (strict latency), rate-critical (specific data rate), and mission-critical (certain degree of reliability) applications [2]. Hence a good Quality of Service (QoS) is essential to stream video over the network and enhancing the Quality of Experience (QoE).

The Internet best-effort service created the lacking deployment of network QoS on Internet architecture since there are problems associated with current used short path routing. First, multiple metrics are hard to use in making routing decisions. Besides, it must consider only optimal cost paths, while cannot exploit the feasible routes. In contrast, the QoS-based routing (QBR) concept appears to provide routing solutions that could achieve the quality needs. The basic idea is to determine network paths to various metrics to supply acceptable QoS for significant application flows, using some knowledge of resource availability in the network, also it must monitor and adapt with QoS parameters variations at network links [3]. In other words, QBR seeks to provide performance guarantees by using multiple metrics in terms of bandwidth, delay, packet loss, etc., as well, it could use best and non-best but acceptable paths [4].

The various metrics used in path calculations could be handled as Single Mixed Metrics (SMM) or as multiple individual metrics [5] [6]. The first one computes and joins different QoS constraints in a single mathematical function, while the second applies many distinct metrics until it finds a feasible path that meets all quality restrictions. In this method, metrics composed by using three different rules, additive (e.g.: delay), multiplicative (e.g.: packet loss), and concave (e.g.: bandwidth). However, the SMM method cannot be sufficient for QoS routing, since it is uncertain that each QoS requirement is respected, as well as the difficulty of defining such function. Moreover, finding a path subject to multiple constraints is non-deterministic polynomial (NP)-complete problem, according to Wang and Crowcroft [5]. In this regard, a NP is a class of problems for which their solution can be found and verified in polynomial time. The algorithm used to solve them is called a non-deterministic algorithm. On the other hand, a problem is assumed to be NP-hard if every problem in NP can be reducible to it in polynomial time. A problem is NP-complete if it is both in NP and NP-hard, which means this problem is the hardest in NP. These types of problems are usually known as intractable. Conversely, the algorithm will be efficient if it can solve the problem in polynomial time. In contrast, if a given problem proves to be equivalent to an NP-complete structure, the solution will require exponential time and it is difficult to solve in polynomial time.

Furthermore, in traditional routing, source-based and hop-by-hop routing algorithms are proposed to achieve QoS routing. Source routing is based on the idea that the flow source is the one who calculates routes on-demand depending on the type of application, and it needs to have information of the entire network that necessary to generate the forwarding paths. On the other hand, hop-by-hop algorithms make routing decisions by using the information available at each router. It allows to reduce the setup delays and distribute the overhead, still, the routers might not be able to avoid the routing loops. However, the existing IP routing does not have the ability to verify the link QoS required to adapt with the requested requirements of QoS for the flow, since it is not aware of the available QoS over the path [4].

The QoS model in the traditional networking faced unresolved issues such as per-hop decisioning, limited network global view, and difficulty of providing different QoS abilities for various application flows. On the other hand, compared to today's architectures, Software-Defined Networks (SDN) appeared to solve the problems of controlling and management of the networks, it decouple control plane and data plane and provide global network view, where controllers can obtain complete topology information and statistics. These features enhance QoS framework by:

- Applying Service Level Agreements (SLAs) and QoS management policies easily by the administrator, that can be changed dynamically at a higher abstraction level, without the need of setting them at each forwarding devices.
- The logically centralized controller, beside maintaining states information for the flow path could facilitate writing quality based routing algorithms and offer end-to-end QoS per flow.

Route decision in QoS routing is made corresponding to traffic characteristic, thus, the OpenFlow networks are a suitable choice, because their forwarding device tables are built on per flow bases. SDN can be very useful to state-dependent routing since the controller can re-route the video traffic through a different path in order to avoid congestion which enhance stream quality at the end receivers side.

This paper proposes a QoS-Based Routing model for video streaming over SDN. It includes the design of an algorithm that selects the optimal path among all paths available in the topology based on collecting the network status to detect the changes in video QoS metrics. Further, the model distinguishes the network traffic by using Differentiated Services Code Point (DSCP), where video flow is marked with the Assured Forwarding (AF) for multimedia streaming service classes, while best-effort flows take the default DSCP.

The rest of this paper is organized as follows. Next section includes the related work. The proposed video QoS-Based routing is described in Section III. The experimental setups and evaluation results are presented in Section IV. Section V depicted a comparison with other proposed QoS-based routing. Finally, conclusion and future work are shown in Section V.

II. RELATED WORK

Well-known techniques have been proposed to offer QoS, such as integrated services (IntServ) [7], differentiated services (DiffServ) [8], and Multiprotocol Label Switching MPLS [9]. However, their implementations are limited due to the decentralized control of the current best-effort networks, resulting in static and complex architecture. Moreover, networking devices are more exposed to failures because of misconfigurations and missing of the automated settings [10].

On the other hand, several SDN routing QoS mechanisms have been proposed to support video streaming. H. E. Egilmez et al. designed the OpenQoS multimedia controller to provide a dynamic end-to-end QoS routing according to the network state [11]. The controller enhances the video quality experienced by end-users by offering two paths. One is a QoS route specified for the multimedia streaming, and the second one is the shortest path for the other data. The route calculation metrics are the delay and the congestion and the available bandwidth. However, the used delay is fixed, which means that the dynamic route is built only on the gathered bandwidth statistics.

The architecture proposed in [10] provides an analytical optimization of the QoS routing problem, which is based on linear programming for scalable video coding (SVC). It aims to minimize the delay and offers no packet loss for QoS traffic. Likewise, [12] and [13] proposed two optimization frameworks for SVC video stream and computed paths using the LARAC algorithm. The first study considers the SVC base layer as lossless flow with no packet loss while the enhancement layers could tolerate packet losses. In case of congestion, only the video base layer allows to reroute to other available non-shortest path routes. The second framework reroutes both base and enhancement layers to QoS-based routes, while the best effort traffic remains on its shortest path. These papers depended on the packet loss rate as a metric to determine the new routes. They assumed that if there is not sufficient available capacity, then, there will be packets loss. However, they do not consider the other possibilities of packet loss, such as security settings or attacks. They did not calculate the actual packet loss rate given by the obtained network state information. The study published in [13] used the delay variation as the constraint with packet loss rate to calculate QoS routes. Also, they ignored the influence of sufficient bandwidth as a routing metric. For video streaming, the delay variation could be improved by increasing the receiver buffers. Whereas, network bandwidth is often more critical for these applications, because if it is limited, this will lead to the queuing delay, and the packet loss rate will be higher, which will decrease the overall QoE of the end user.

Jinyao et al. [14] presented the HiQoS SDN framework to guarantee QoS. It is based on differentiated services and computes multiple paths between the source and destination. The framework distinguishes between different types of traffic by source IP address and categorizes user traffic into a video stream, interactive audio/video, and best-effort flow. Additionally, a modified Dijkstra algorithm was used to select the optimal path for the new flow according to the lowest bandwidth consumption. Using multiple paths enables HiQoS to be resistant to link failures through rerouting flow to another path. However, it only used the minimal bandwidth utilization of a queue as a metric, ignoring other video streaming quality

metrics. Also, it classifies the flow with the source IP address, this might be not precise for video streaming, since different videos with different quality demands may be originated from the same device.

ARVS [15] suggested a two-level QoS routing approach for video streaming over SDN. They used the delay variation and cost to select the optimal paths for videos. Their work is like [13], except they utilized the shortest plus feasible QoS paths for all the traffic types. Both, the base layer video stream (level-1 QoS) and best-effort traffic, routed through the shortest path. ARVS will check this path against jitter constrain periodically, if it has jitter values over a threshold, the available bandwidth along the QoS path is examined, in case that the bandwidth is sufficient, the level-1 flow has priority to be re-routed to this path, and enhancement layer packets (level-2 QoS) remained on the shortest path, while the level-2 re-routes to the QoS path and the base layer packets stay in the shortest path. However, packet loss is the most important constrained metric for a video stream at the network level and buffers at the application level can solve the jitter problem.

In the same direction, [16] offered a QoS-enabled management framework to support a queuing mechanism and route optimization algorithm to guarantee the transmission of the flow over SDN networks. It provides a suitable QoS for video streaming and multimedia applications, and also classifies the flows into different QoS levels. Their algorithm solves the constrained shortest path (CSP) problem based on the delay parameter. Besides, they compute the routing path cost with two additive metrics (delay and packet loss).

In 2017, authors of paper [17] presented a routing optimization system for SDN by applying reinforcement learning (RL) to enhance network QoS. The proposed routing protocol used this artificial intelligence (AI) method to select the optimal paths with the least cost in terms of delay, loss rate, and bandwidth. Additionally, Guo et al [18] applied reinforcement learning (RL) in hybrid SDNs. It achieved link load balancing with the avoidance of routing loops by responding to dynamically changing traffic. Moreover, in [19], authors used a deep reinforcement learning (DRL) scheme to optimize routing in SDN of data-center networks. The method combines different network resources, such as bandwidth and cache memory, in order to minimize the delay. Then, it uses this information to improve the routing performance.

Rego et al. [20] analyzed the effect of OSPF protocol on the network quality. Their study compared the protocol performance over SDN and traditional network. Another work, [21], took advantage of the SDN and AI to propose a dynamic routing metrics calculation for multimedia data. The presented proposal modified the OSPF protocol equation considering several metrics (bandwidth, delay, and loss). Besides, they implemented a messages exchange protocol between controller and switches to adapt the metrics according to the current topology state. Their results show that the delay and packet loss decreased, while bandwidth utilization increased.

Also a performance enhancement framework for IP video surveillance (IPVS) is presented in [22]. It takes advantage of SDN to adjust QoE bitrates and reroute traffic to maximize the resources available on the network. Employing rerouting ensures choosing the shortest and less loaded paths, so the video destination received the high-priority streams at superior quality. Framework results minimize packet loss, jitter, and latency, while it is also optimized the throughput.

Yamansavascular et al. [23] proposed a dynamic fault tolerance solution to improve the QoE of video streaming. They detect the congestion in the SDN link layer by using bidirectional forwarding detection (BFD) protocol, then, they employ data plane link failure tolerance mechanism to find alternative paths in order to offer good QoE.

Most of the previously mentioned studies focused on the CSP problem that tries to find the least cost path which satisfies only one constraint, while ignoring the multiple constrained shortest path (MCSP) problem. MCSP problem aims to find a path subject to multiple constraints. Video flow is one class of service that requires combinations of routing metrics to realize its QoS. Therefore, CSP does not estimate or guarantee that the path could fulfill all QoS constraints individually.

III. VIDEO QoS-BASED ROUTING SCHEME

This section presents the main parts of the proposed video QoS-based routing approach.

A. Route Constraint Metrics Selection

Video Streaming QoS requirements have to map into path metrics because the network should guarantee QoS relying on the definition of the proper metrics. In general, finding the QBR path depends on many constraints such as additive (delay and jitter), multiplicative (packet loss), and concave (bandwidth) metrics. However, the computing complexity of choosing the best route can be very high, depending on the number of used metrics and their sequence. Wang and Crowcroft [5] proved that finding a path subject to combinations of additive and multiplicative metrics is NP-complete if the number of metrics equals two or more metrics. Therefore, the choice of metrics and their count will depend on the importance of those metrics for the type of application (video streaming).

This work is concentrated on enhancing the QoE of two types of video streaming resolutions: High Definition (HD) and Stander Definition (SD) video resolutions. Therefore, we investigated the effect of different video QoS parameters that can influence the QoE and we focused on network-level related QoS parameters rather than video application-level parameters.

Video streaming is one of the most bandwidth demanding services, where its demands increases with the resolution of the video content. As a result, the bandwidth is more significant for video streaming, especially because higher video resolution means higher bitrate. Thus the network has to provide more capacity. Further, insufficient bandwidth will increase the delay and loss rate, leading to a decrease of end-user QoE. With high encoding bit rates, the stream is more exposed to packet loss, which has severe impact on the video streaming QoE. Users could experience frame freezing, complete loss of the video, or others problems depending on what video frames are lost. According to [24] and [25], the packet loss and bandwidth have a

more effect on video streaming than jitter and delay. This happens because the jitter could be avoided by implementing de-jitter buffers, or by the playout buffers in the video application at the receiver side. Consequently, we applied packet loss and bandwidth as path metrics in our QoS-based routing method. Then, the impact of these QoS parameters on the quality of HD and SD video resolutions has been studied.

OpenFlow port counters are used to calculate the packet loss and bandwidth on each link. The controller obtains the switch port counters by sending the “OFPortStatsRequest” message. Then it receives the “OFPortStatsReply”, which contains the received byte count and other statistical data.

B. QoS Routing Problem Formulation

This study modeled the SDN network as a graph $G = (V, E)$, where V is a set of nodes, and $E = \{(i, j) : i, j \in V\}$ is a set of links. The path or route metrics are the packet loss rate ($PLR(r)$) and the available bandwidth ($\mathcal{A}(r)$). In order to obtain the $PLR(r)$ of the OpenFlow network, several factors should be considered, such as the OpenFlow port type if it is output source or destination ingress port, and the TX and RX buffers which are used to transmit and receive data. Depending on these factors, the PLR equation calculated based on TX buffers within output ports, and RX buffers with ingress ports, loss rate metric is associated with the specific link (i, j) between two network points, node i and node j , where $i = 1, \dots, n - 1$ and $j = i + 1, \dots, n$, and it is defined by:

$$PLR_{i,j} = \frac{(N_i TX_{outport} - N_j RX_{inport})}{N_i TX_{outport}} \times 100 \quad (1)$$

Where N is number of port, RX_{inport} and $TX_{outport}$ are the set of buffers at ingress port and output port interface. Then the path loss probability between node s and node t approximated as:

$$PLR(r_{s,t}) = 1 - ((1 - PLR_{s,i}) \times (1 - PLR_{i,j}) \times (1 - PLR_{j,k}) \times \dots \times (1 - PLR_{l,t})) \approx$$

$$PLR(r_{s,t}) = PLR_{s,i} + PLR_{i,j} + PLR_{j,k} + \dots + PLR_{l,t} \quad (2)$$

Additionally, in order to compute available bandwidth metrics through an end-to-end path between source and destination, this work needs to find the bandwidth utilization $\mathcal{BU}_{i,j}$ and available bandwidth $\mathcal{AB}_{i,j}$ for each link (i, j) between forwarding devices i and j on the topology. Also, this study assumed that the maximum capacity of every link in network topology $C_{i,j}$ is known, and the controller will estimate the link utilization after gathering network topology statistics. So, to find the bandwidth utilization, we suppose that the OpenFlow controller received OFPortStatsReply message at a time (t_1), which contains the received (RB_{t_1}) bytes, and, after a period of interval (ΔT), a second OFPortStatsReply received at the time (t_2) which has (RB_{t_2}) bytes, the $\mathcal{BU}_{i,j}$ measured by bit/s calculated as follows:

$$\mathcal{BU}_{i,j} = \frac{RB_{t_2} - RB_{t_1}}{\Delta T} \times 8 \quad (3)$$

Then, the available bandwidth for the link (i, j) is computed as:

$$\mathcal{AB}_{i,j} = C_{i,j} - \mathcal{BU}_{i,j} \quad (4)$$

To sum up, the minimum unoccupied link bandwidth among all links (i, j) along that path determine the end-to-end path available bandwidth $\mathcal{A}(r)$. This link is known as the bottleneck, and its bandwidth is considered as the maximum possible bandwidth for the path, also recognized as the width path. $\mathcal{A}(r)$ for a path between node s and node t defined by the following equation:

$$\mathcal{A}(r_{s,t}) = \min [\mathcal{AB}_{s,i}, \mathcal{AB}_{i,j}, \mathcal{AB}_{j,k}, \dots, \mathcal{AB}_{l,t}] \quad (5)$$

Finally, if given two constraints, \mathcal{BT} (bandwidth threshold), and \mathcal{PLT} (packet loss threshold), and there is a source, s , and a destination, $t \in V$, the routes between them are represented as a set of $\mathcal{R}(s, t)$. Then the QoS routing problem is to find a feasible route $r \in \mathcal{R}$, joined s and t , where:

$$\mathcal{A}(r_{s,t}) \geq \mathcal{BT} \text{ and } PLR(r_{s,t}) \leq \mathcal{PLT} \quad (6)$$

C. Processes Description

By using terminologies defined in the previously, this research proposes a flow-based routing strategy for video streaming traffic subject to meet various QoS requirements. The primary goal of the routing strategy described below emphasizes to meet two routing constraints and efficiently determines two feasible paths for each video streaming flow on the given current state of the network. Fig. 1 demonstrates the general paradigm of the QoS-based routing.

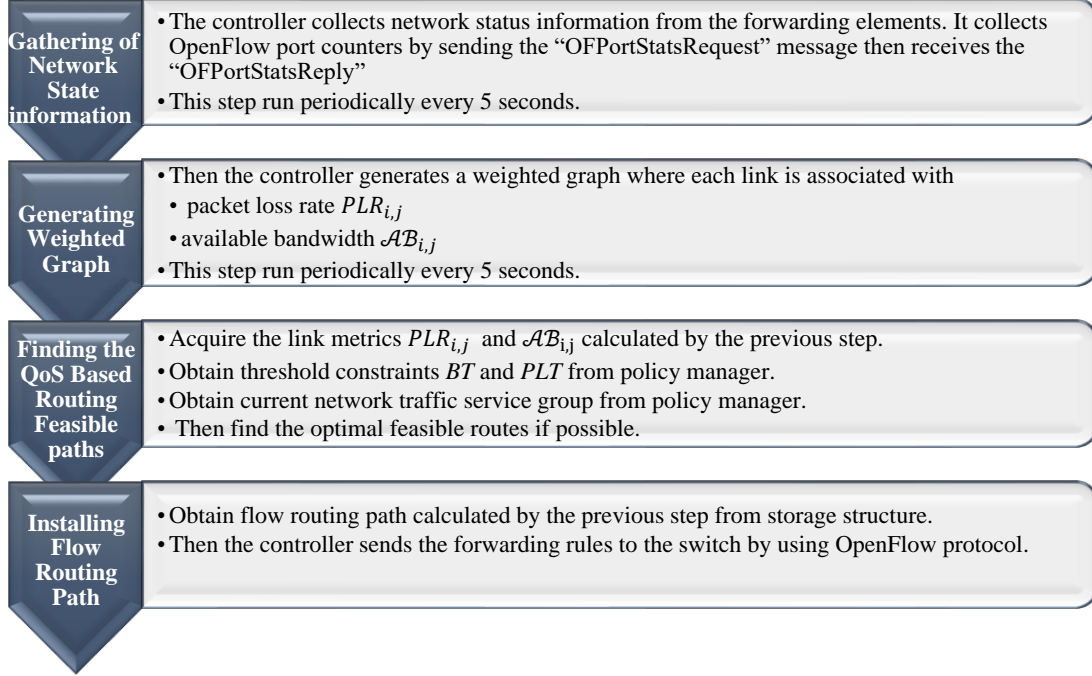


Fig. 1. Video QoS-Based Routing Steps

D. Proposed Routing Algorithm

Next we present the design of the routing algorithm in order to solve the QoS-based routing problem and achieve the optimization goal. There are two main ideas behind the design:

- The optimization criteria for the video streaming in this work is to minimize packet loss by choosing a path with lower end-to-end packet loss rate, but it can not exceed a specified threshold. Plus, maximize the capacity by choosing a path with high available bandwidth that meets video streaming bandwidth boundaries.
- The precedence between metrics is an important factor because it reflects the influence of network QoS parameters on overall QoE. Therefore, the proposed algorithm takes the bandwidth as secondary metric, and loss as primary one, since the latter is more impairing for video streaming, as it has been analyzed earlier, and there could be a route with high capacity but it can be losing the packets by different causes than congestion.

As discussed earlier, solving QoS problem depends on using multiple metrics to be NP-complete. This work proposes an heuristic algorithm based on the source routing algorithm presented by Wang et al. [5]. We considered it because of two features. Firstly, it computes forwarding paths on demand per flow basis, which is very appropriate for video streaming conditions. Secondly, in order to install the routing path, the entire network topology must be recognized, and this characteristic is offered by the SDN controller where it can access the full routing information of each link needed for the path computation. Our approach also includes in the algorithm the packet loss rather than delay, further finding two feasible paths based on k-short paths (KSP) Yen, Jin Y. [26] rather than only Dijkstra [27]. In addition, it can find the widest path based on the bandwidth metric only; if the type of QoS policy permits it. In this regard, we have defined a policy rule to determine which traffic receives the QoS. It states that the network traffic must belong to one of three service groups: restricted QoS constraints called (group A), tolerable or soft QoS constraints (group B), and best-effort (group C). The flows in group A need enforcement of their QoS constraints, and those in group B are tolerant of acceptable performance guarantees, while group C does not claim any QoS guarantees. The proposed algorithm is called Two Lowest Loss -Widest Paths Algorithm (TwoLLWPs). It works as follows:

- There are two phases, elimination and search. First, all edges on original network topology graph that do not meet the bandwidth threshold are pruned. So, any paths in the resulting graph G satisfies $\mathcal{A}(r) \geq BT$. second, when there is more than one widest path meeting QoS bandwidth requirement, the algorithm begins to search two routing paths from

the source node to the destination with the lowest packet loss rate $PLR(r) \leq PLT$ if it possible based on Dijkstra algorithm and Yen's k-shortest path. The two paths must meet packet loss rate restriction. The path with lowest packet loss rate is called the lowest loss-widest-first path, while the second is called the lowest loss-widest-second alternative path.

- If at least one path is found, the flow admission status is changed to admitted, otherwise it is changed to "rejected".
- Furthermore, the algorithm does not have to find the minimum loss rate paths to all nodes, rather it finishes either when the destination node t is permanently identified or when the packet loss rate exceeds the threshold before reaching t .
- Yen's algorithm is used to determine the first k-shortest path by using Dijkstra to find the shortest path between two nodes. Then it starts to determine all other k-shortest paths. The TwoLLWPs modifies Yen's algorithm by removing the first call of Dijkstra, because the first path is already computed.
- If it fails to find a feasible path meeting both, loss and bandwidth requirements, the algorithm returns the widest path based on bandwidth constraint by calling the Widest_Path_Dijkstra; if the policy rule associated with the flow allows this, because the biggest available bandwidth will be more desirable metric for video traffic rather than shortest path. The widest path is computed in one condition, if the application flows belong to group B, which has tolerable restrictions.

Fig. 2 shows the flowchart algorithm of process described before. Besides, all the procedures mentioned above are achieved by implementing these steps in Algorithm 1.

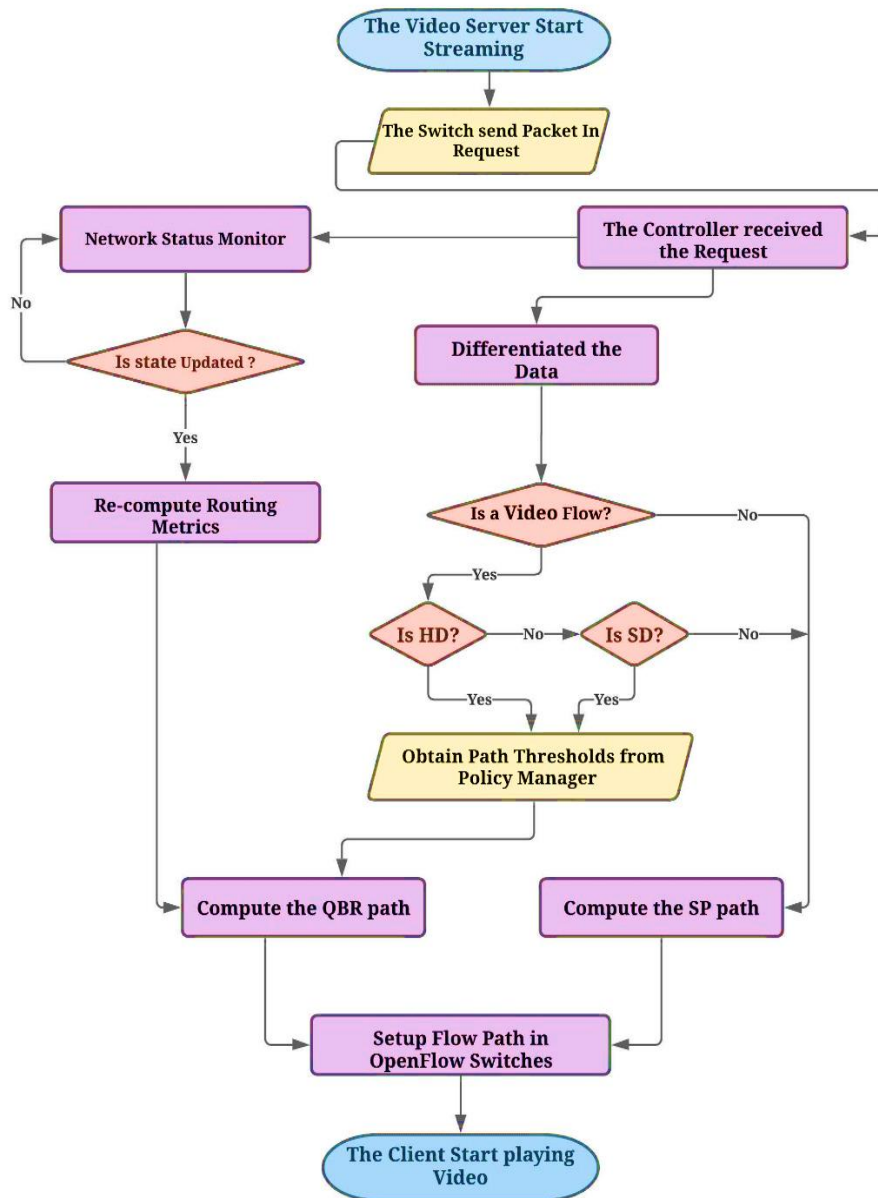


Fig. 2. Algorithm Flow-Chart

Algorithm 1: Two Lowest Loss - Widest Paths (TwoLLWPs)

Input : Weighed $G = (V, E)$, a graph with node and edge set, each edge has two weight values
the $PLR_{i,j}$ and $\mathcal{AB}_{i,j}$, values are non-negatives
A source $s \in V$, a destination $t \in E$.
Two constraints \mathcal{BT} and \mathcal{PLT} .
Flow Group one of this values $FG : [A \text{ or } B]$.

Output : HashMap with: paths set, path type, path counts and Flow Admission Status: FAS : [A for Admitted, R for Rejected].

Step 1 : To create $newG$, $\forall i, j : PLR_{i,j} = \infty$ if $\mathcal{AB}_{i,j} < \mathcal{BT}$.
// Prune phase

Step 2 : $SPT = \{s\}$; $\forall i \neq s PLR(p_i)^* = PLR(p_{s,i})$,
 $parents = \{\}$. //initialization

Step 3 : Find $k \notin SPT$ so that $PLR_k = \min_{i \notin SPT} PLR(p_i)^*$.
If $PLR_k > \mathcal{PLT}$, { //No path could be found
If $GF == B$, {
 $r^* = Widest_Path_Dijkstra(G, s, t, \mathcal{BT})$ If $r^* \neq \emptyset$,
 $Build_HashMap(r^*)$
Else $FAS = R$.
}
 $FAS = R$.
}, **End Algorithm.**
If $t \in SPT$, { //At least one path is found
 $FAS = A$.
 $r = Build_Path(parents, t)$
 $KSP = YenKSP(newG, A[0]=r, k=2)$ // Call yen's algorithm to find second path
If $KSPcount \geq 1$, $Build_HashMap(KSP)$
}, **End Algorithm.**
 $SPT := SPT \cup \{k\}$.

Step 4 : $\forall i \notin SPT : PLR(p_i)^* := \min [PLR(p_i)^*$
 $, PLR_k + PLR_{k,i}]$,
 $parents[i] = k$.

Step 5 : Go to Step 3.

The time complexity of this algorithm depends on its phases. The first phase prunes the graph by searching through all vertices and edges to eliminate those links that do not meet bandwidth constraints via replacing their packet loss rate by ∞ . This step is supposed to be executed once and requires $O(N^2)$ run time. Despite this, the floodlight controller already has a function to generate a weighted graph, so we modified it by inserting the “if” condition to keep only the links that fulfilled the bandwidth constraint. So, the graph is created from the beginning with only the desirable links. The second phase is to find two paths using Dijkstra and Yen algorithms. We implemented Dijkstra using a Fibonacci heap priority queue so it will

run in $O(M + N \log N)$ time rather than $O(N^2)$, consequently Yen's algorithm makes KN invocations of Dijkstra's thus it takes $O(KN(M + N \log N))$ [28][29]. The proposed algorithm set $k = 2$, as a result, the complexity of using two algorithms gives equation 2:

$$\text{Complexity} = O(2N(M + N \log N)) \quad (7)$$

Fig. 3 shows the messages exchanged between the OpenFlow switch, the SDN controller and the proposed QoS module. The message flow diagram includes the initial negotiation and the major messages between OpenFlow switch and the SDN controller. Whenever a server starts streaming, the switch sends a packet_In to the SDN controller. Then, it passes this request to the QoS module, which begins to set up an optimal path for video steaming.

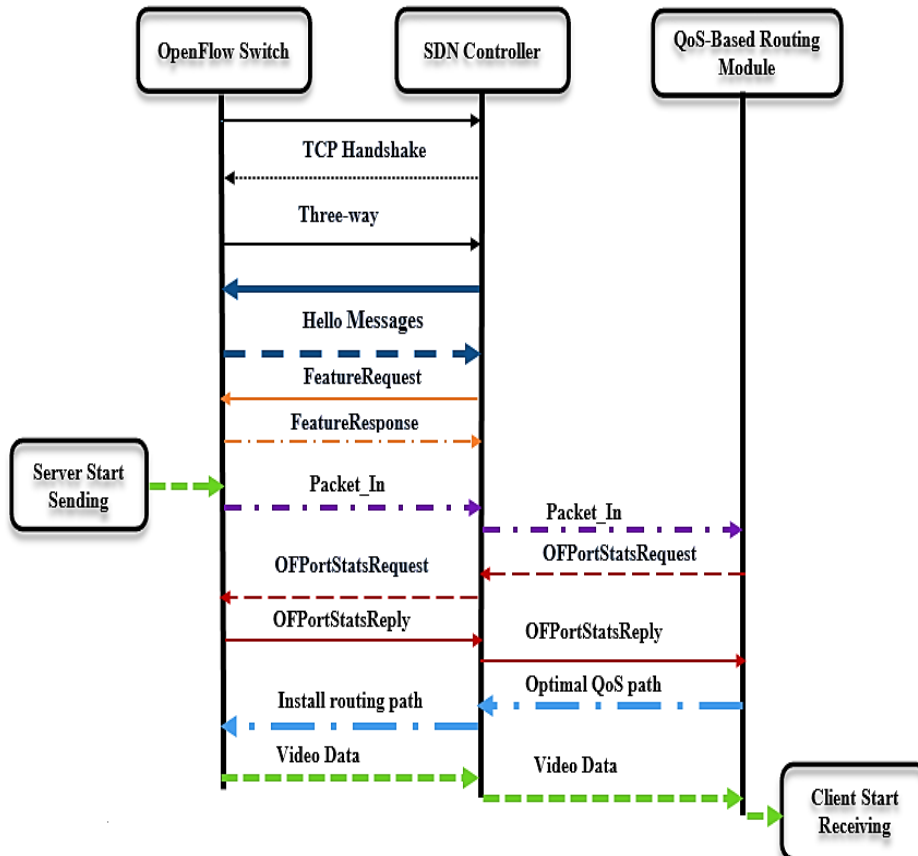


Fig. 3. Message Flow Diagram

IV. EXPERIMENTAL RESULTS AND ANALYSIS

This part presents the experimental assessment undertaken to evaluate the proposed solution. Different scenarios were designed to study its performance. The simulation testbed environment and the parameters are described. Additionally, the evaluation results are analyzed and discussed.

A. Testbed Setup

Table I describes the network parameters used for the experiments.

TABLE I. : GENERAL NETWORK PARAMETERS

Topology Parameters	Values
Network links packet loss rates (%)	Ranging between: 0.001, 0.005, 0.05, 0.5, 1.0, and 2.0
Link bandwidth	Ranging between: 100 Mb/s, 10 Mb/s and 1 Mb/s
Number of links	22
Number of Nodes	1 Controller 9 Switch 6 Hosts (one of them is the Server)

Simulation was performed using Mininet emulator. We implemented the network topology shown in Fig. 4. This figure is designed by SmartDraw software [30].

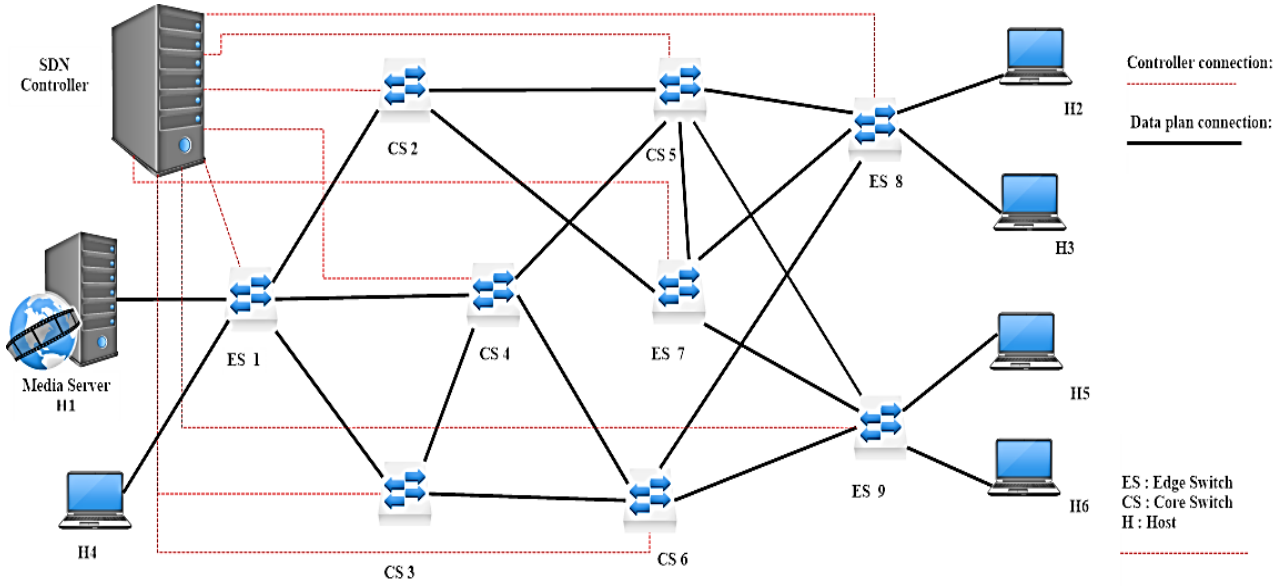


Fig. 4. General Perspective of Topology Design

The test was conducted using two laptops. One with Intel core™ i7-7500U, 2.70 GHz CPU, 2 Cores and 16 GB RAM. The other laptop with Intel core™ i7-4600U, 2.10 GHz, 2 Cores and 8 GB RAM. Also the setup included two virtual machines with two processors installed on Oracle VirtualBox Graphical User Interface Version 6.1.10. One of them contains the Floodlight controller master version and the second contains the Mininet Simulator version 2.3.0. Table II summarizes the used software and hardware specifications in both laptops

TABLE II. : USED HARDWARE AND SOFTWARE

Hardware or Software	Specifications
Ethernet Link Bandwidth	100 Mb/s
Computer OS Specifications	Microsoft Windows 10 Home , x64-based PC System and Linux Ubuntu 18.04.5 LTS (Bionic Beaver) OS, x64-based PC System
Programming Languages and Editors	<ul style="list-style-type: none"> • Java JDK 8. • Python v3.8. • Eclipse Editor. • PyCharm Editor.
Video System and Measurement Tools:	<ul style="list-style-type: none"> • GStreamer v1.18.4. • VLC Player. • Wireshark packet analyzer. • MSU Video Quality Measurement Tool 12.0 beta.

B. Measurement of QoE Metrics

Generally, video streaming QoE can be measured by subjective or objective methods, or both of them. Subjective metrics are conducted by asking subjects to rate the video they have been watching, the subjects rate the quality by excellent, good, fair, poor or bad. The Mean Opinion Score (MOS) includes well-known QoE scales, which is an average of scores across subjects, the MOS maps the ratings between excellent and bad to numerical scores between 5 to 1 [31] [32].

The full reference (FR) model is used as an estimation method in video streaming QoE. It works by accessing both the original, called reference, and processed video, called distorted, to assess the quality. The properties of the two tested videos are compared frame-by-frame to check different features such as contrast features, and color processing. We use two types of FR measurement metrics for QoE estimation. The Structural Similarity Index Metric (SSIM) [33] and the Video Multimethod Assessment Fusion (VMAF) which has been recently proposed by Netflix [34]. Table III shows an approximation objective QoE scales for SSIM and VMAF video quality metrics mapped to the MOS scores [35] [36] [37].

TABLE III. SSIM AND VMAF VALUES MAPPED ACCORDING TO MOS SCORES.

MOS	SSIM	VMAF
Excellent = 5	> 0.99	80 - 100
Good = 4	$\geq 0.95 \ \& \ < 0.99$	60 - 79
Fair = 3	$\geq 0.88 \ \& \ < 0.95$	40 - 59
Poor = 2	$\geq 0.5 \ \& \ < 0.88$	20 - 39
Bad = 1	< 0.5	< 20

C. Proposed Algorithm Testing Scenarios

Three test cases were designed to study the TwoLLWPs algorithm. It allows us to explain the importance of each QoS parameter and its impact. Their goals are as follows. The first test case compared the floodlight default mode (FDM) with video QoS mode (VQM), where in FDM all data traffic is routed by the original floodlight shortest path algorithm. In contrast, VQM transmits two types of data traffic, videos (combining two types, SD and HD) and best-effort. The video routed through the generated paths by TwoLLWPs algorithm, while the best-effort data remain at floodlight shortest paths. Second test case studies the various packet loss rate constraints influence on the video streaming QoE. Finally, third test case examines impact of different bandwidth constraints.

1) FDM vs VQM Test Case.

The SDN network Link bandwidth used in this test is 100 Mb/s, and the links of the longest paths have no loss, while other links take value > 0 and $\leq 2\%$ loss metric. There is a background traffic that was generated using iperf testing tool between host 4 and host 6 (UDP traffic at 10 Mb/s). The test video samples used are the Sintel Trailer, obtained from Sintel website copyright under Blender Foundation [38], and Caminandes videos under the Creative Commons Attribution 3.0 license from the caminandes.com [39]. The policy rule categorizes the flow as belonging to the group (A). QoE and QoS parameters and their values are summarized in tables IV and V.

TABLE IV. VIDEO QOE PARAMETERS

Video Name	Type SD or HD	Bitrate kbps	Size MB	Duration	Frame /s
Sintel Trailer 480	SD	540	4.16	52 s	24
Sintel Trailer 1080	HD	2116	13.9	52 s	24
Caminandes_llamigos_1080p	HD	1042	191	2:30 m	24

TABLE V. NETWORK QOS PARAMETERS

Video Type	FDM	VQM	
		Packet Loss Rates Constrain Metrics	Bandwidth Constrain Metric
Class 28 (SD)	-	0.5%	3 Mb/s
Class 30 (HD)	-	0.005%	10 Mb/s

Fig. 5 shows the variance between using FDM and the proposed video QoS-based routing. The fact of missing frames has a significant influence on FDM, which results in bad viewing experience compared to VQM because of using the shortest path rather than QoS routing. The video artifacts problems seen in FDM include annoyance and blocking in frame pixels. Also, sometimes complete image losses are shown. The frame in VQM is relative to the original frame with few color distortions.

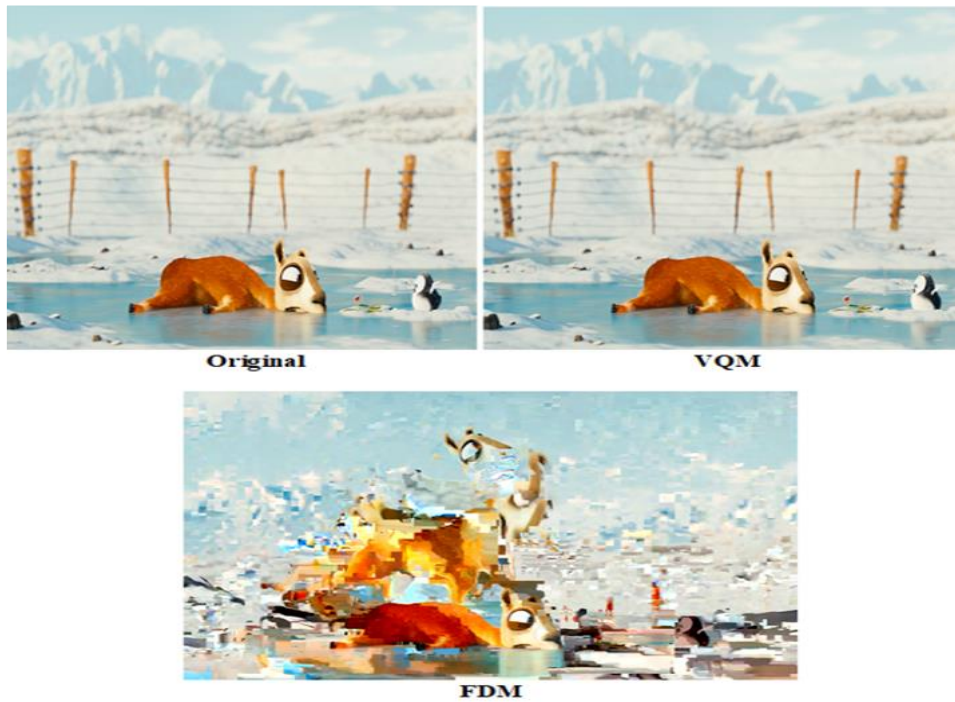


Fig. 5. Original Video vs FDM and VQM

Fig. 6 and Fig. 7 show the impact of the TwoLLWPs and FDM on different received video sequences resolutions. X-axis represents tested frames and the Y-axis represents the structural similarity index metric (SSIM). As observed, the video QoE is affected by QoS network parameters. In this experiment, the video without the QoS-based routing scheme reflected a dropping in the perceived video quality. For SD type, the obtained SSIM values for Sintel (480) video indicate higher quality viewing behavior. In addition, HD performance results also increased when it is compared to shortest path routing.

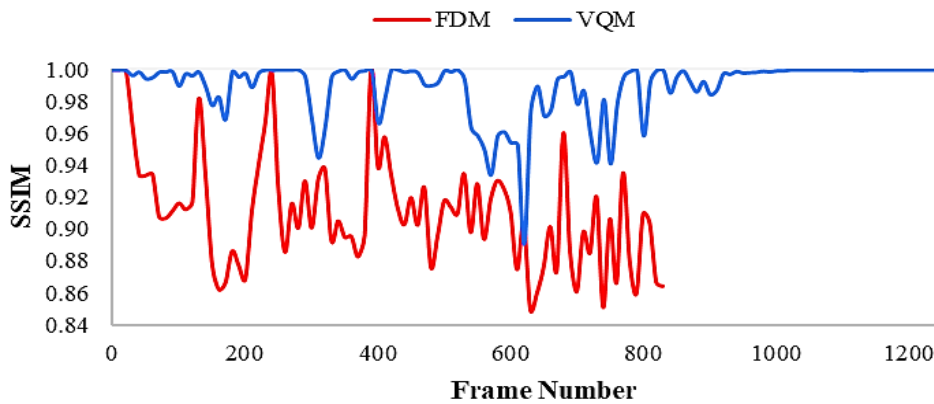


Fig. 6. FDM vs VQM impact on SD Videos

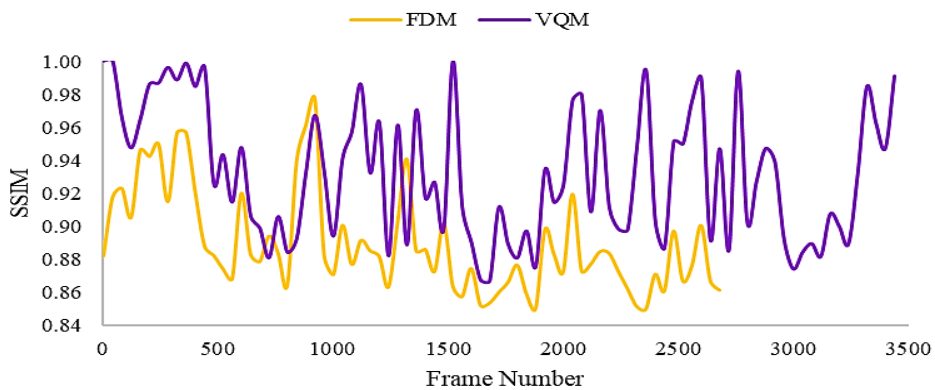


Fig. 7. FDM vs VQM impact on HD Videos

In addition, Fig. 8 displays the mean VMAF scores for Sintel Trailer samples. The results exhibit that the proposed approach offers high scores, which indicates good viewing quality for SD and fair for HD. While FDM's highest average value does not exceed 30 for SD, for HD value is less than 10, which suggests a poor, and bad QoE perception.

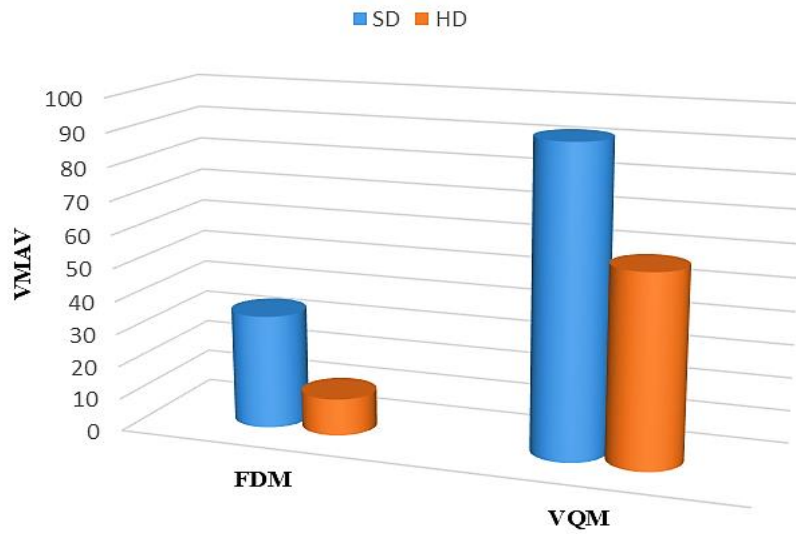


Fig. 8. Average values of VMAF Objective Quality Metric

Also, we investigated the network throughput achieved in the case of using FDM and VQM. This time we used the Caminandes Llamigos video (with a resolution of 1920 x 1080) and Wireshark packet analyzer. Fig. 9 presents the video traffic throughput for FDM, and Fig. 10 shows the network throughput received for the proposed model. As noted in the first figure, the throughput decreases significantly due to the packet loss increase, where the maximum value reaches only 2,500 bytes per second. Conversely, the second figure shows that VQM achieves a higher throughput, where the maximum bytes per second range between 400,000 and 500,000; the reason is that the TwoLLWPs reduces the packet loss probability and gives information of the available bandwidth, which improves the network throughput.

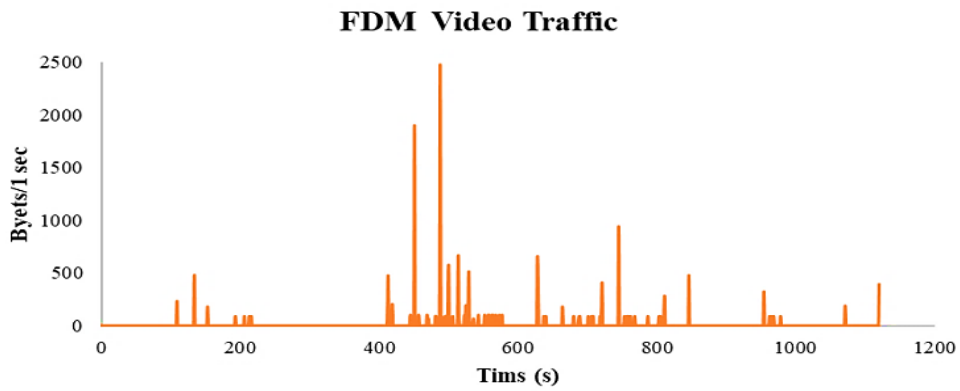


Fig. 9. FDM Network Throughput

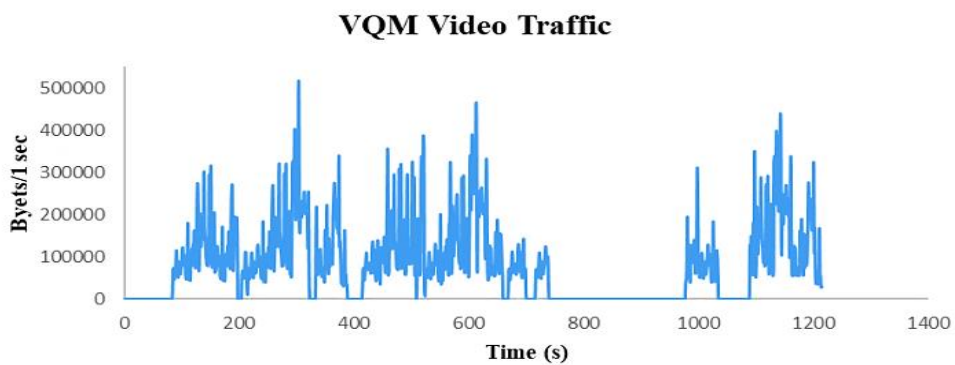


Fig. 10. VQM Network Throughput

Many remarks can be summarized from the above results. For example, the number of video frames received by VQM is higher than the number of frames received by FDM because the lossy links have been avoided with the proposed QoS-based routing algorithm. Moreover, the HD video resolution produces lower objective SSIM and VMAF metrics ratings in comparison to SD resolution, because HD frames contain more information than SD, therefore in case of packet loss, the HD incurs more degradation in the video viewing experience.

2) Influence of Packet Loss Rate as Constraint.

This test seeks to study the effect of different packet loss values as constraints on the proposed routing algorithm. The parameters deployed in the experimental set-up are the following ones. The video samples used for this test were Sintel Trailer SD and HD, and the policy rule categorizes the flow as belonging to the group (B). Link capacity is set to 10 Mb/s. The links of the longest paths are divided into two parts: (i) part with no loss, and (ii) part with 0.3% loss, while other take value > 0 and $<$ loss metric. The background traffic is set to be about 3Mb/s. The loss rates constraint metrics used for this test are: 0.001, 0.005, 0.05, 0.5, 1.0, and 2.0, while the bandwidth constraints metrics are fixed to 5 Mb/s for SD and 10 Mb/s for HD. For the FDM, the metrics settings in floodlight have no effect. Finally, the video quality was measured by SSIM.

Fig. 11 presents a sample of received HD video by the TwoLLWPs with different loss rates metrics. FDM is compared with the original video. The results suggest that the QoE of the coded video increased when using the proposed algorithm. As noticed, even with metric up to 2% PLR, the perceived impairments are less than the default shortest path .



Fig. 11. Impact of setting the TwoLLWPs by packet loss rates (0.005% or 2%) against FDM.

Fig. 12, Fig. 13, and Fig. 14 show the impact of different packet loss rate values on SD and HD videos. Fig. 12 shows that when the PLR=0.05%, SSIM decreases. A similar result happens when PLR=1%. When PLR=0.5%, the video has higher SSIM values. The degradation of quality when we use small loss rate values is due to the probability of rejection by the algorithm because it takes more time to find feasible paths. If it fails to find one, it takes the widest path since the policy rule used in this test configures the flow to belong to group B, which has soft constraints, and the widest is considered acceptable for this group. However, this reduced SSIM. In contrast, higher loss rates initially result in higher scores as the probability of finding paths meeting the two constraints increases. However, the video quality significantly improves after a while when the controller reads the new network states.

In a similar manner to SD video, Fig. 13 presents the SSIM values measured during HD video delivery, when the PLR=0.005%, the SSIM decreases at the beginning but improves later, and when PLR=0.5%, SSIM scores are higher at the start of video reception but, it dropped afterward. It is explained using the same reasons aforementioned in the analysis of SD results. Moreover, when the packet loss rate is high, it indicates a degraded received QoS, e.g., when PLR=1%, PLR=2%. Finally, Fig. 14 shows that when the values of packet loss rate metrics are so close to each other, there is not much difference in the observed quality. Accordingly, as can be seen from the figures, choosing a loss metric that is not too small will enhance the performance of the routing algorithm; this must, however, be in balance with selecting the appropriate threshold.

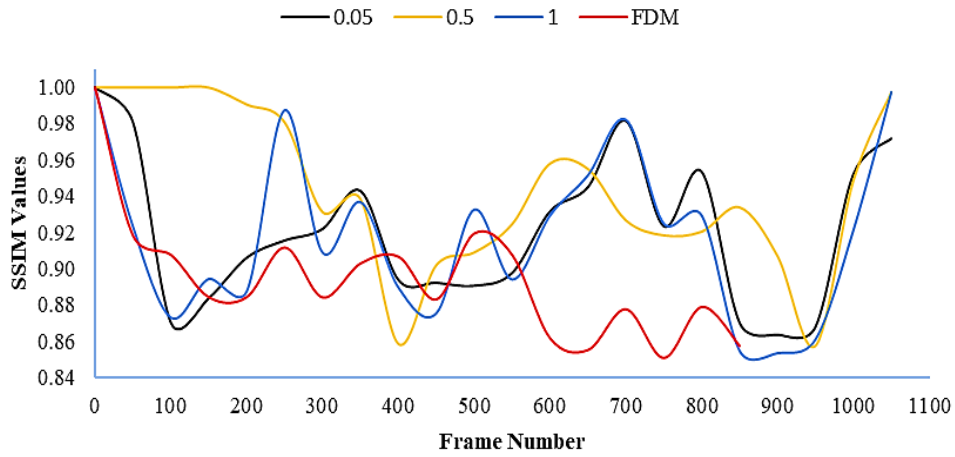


Fig. 12. Impact of many PLR values on QoE of SD video

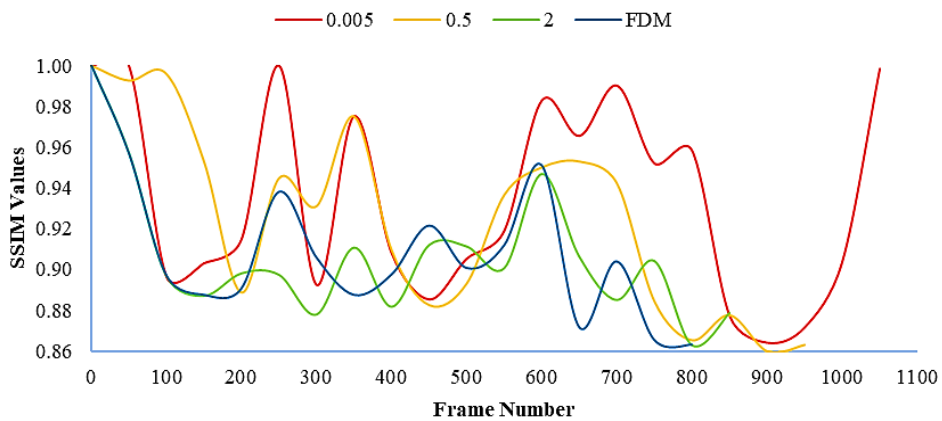


Fig. 13. Impact of a Number of PLR on QoE of HD video

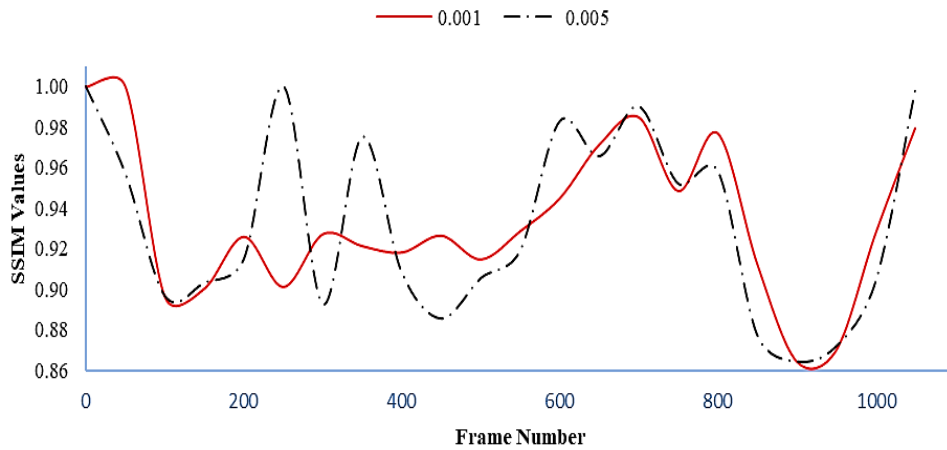


Fig. 14. Impact of Several PLR (0.001 and 0.005) on QoE of HD video

3) Influence of Bandwidth as Constraint

In order to evaluate bandwidth metric impact on the performance of the routing algorithm, different bandwidths have been applied (1, 5, 10 and 20 Mb/s). The link capacity set to 50 Mb/s, and two hosts generate about 20% UDP best-effort-traffic. The longest paths have no loss and the rest take values > 0 and $\leq 2\%$, and the loss input metrics are 0.5% and 0.05% for SD and HD respectively.

Fig. 15 presents the obtained results. It shows that badwidth constraints strongly affects the videos. As we can see, in Fig. 15, the differences in the results are irregular; e.g., in SD, the average SSIM of 10 Mb/s obtains better results than that of 20 Mb/s, despite that the 20 Mb/s means the chosen path is supposed to have a rate higher than 20 Mb/s. We can explain the behavior by the fact that the selected route ultimately depends on the loss metric; for example, the admitted path rate for the

threshold=10 might be 40 Mb/s, and for the threshold=20, it might be 22 Mb/s. As well, the same observations are given in the case of HD videos. Moreover, due to the best effort that competes for resources, the quality of videos may have dropped since QoS-based routing lacks a mechanism to reserve resources. Finally, small bandwidth metrics increase the options for routing paths, while high bandwidth metrics reduce these options and increase rejection rates.

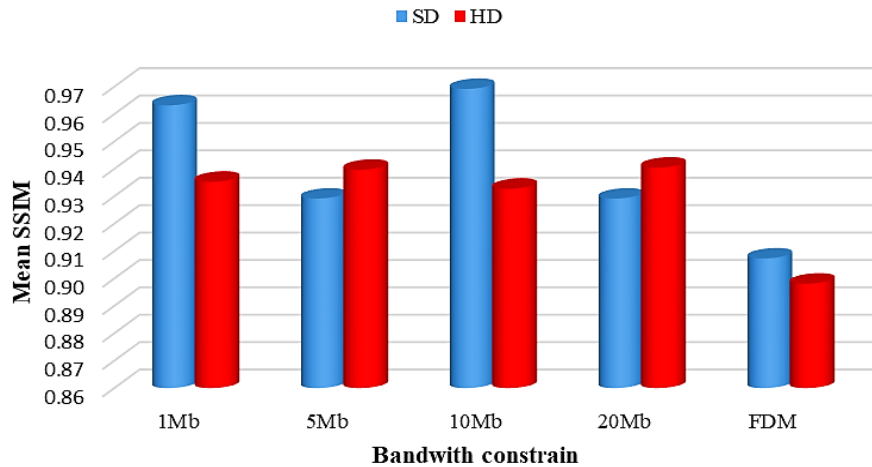


Fig. 15. Impact of several bandwidths on QoE of SD and HD Video

V. DISCUSSION

Table VI illustrates a comparison between the proposed approach and other QoS-based routing proposals. The majority of the studies are focused on achieving only one constraint of video streaming [10], [13], [15], and [16], while some others are focused on optimizing the performance without addressing specific QoS requirements [14] and [21]. Similarly, many of them use the cost of route selection based on a single mixed metric, and, as we stated previously, when it depends on the SMM, it does not guarantee each QoS parameter individually. Alternatively, our proposal, TwoLLWPs, meets multiple individual QoS metrics.

Moreover, the effective way to evaluate the quality of the video is by considering the user perspective and assessment, which is known as QoE. It is achieved by implementing subjective tests or by objective QoE metrics. However, three studies ([14], [16], and [21]) evaluated the video quality based on network-level performance measurements. Additionally, [10], [13], and [15] measured the video quality via Signal-to-Noise Ratio (PSNR), but it does not reflect the human perception. For this reason, it cannot be considered a reliable method for assessing QoE [40]. Instead, we use SSIM and VMAF, which can analyze blurring, luminance, contrast, global noise, as well as blocking and color distortions to detect artifacts that can be perceived by human eyes.

TABLE VI. COMPARISON OF RELATED WORK WITH THE PROPOSED SOLUTION ON QoS-BASED ROUTING.

#	QoS-Based Routing Algorithm	Routing Metrics	The applied Constraints	Video Parameters	QoS/QoE Measurement metrics
Civanlar [10]	The Solution not Developed	Available Bandwidth	Delay	SVC B/E layer	PSNR
Egilmez [13]	LARAC	Two mixed metrics: Packet loss and jitter	Jitter	SVC B/E layer	PSNR
Jinyao [14]	Modified Dijkstra	Two mixed metrics: Delay and Bandwidth	Minimal Bandwidth Utilization	N/D	Server Response Time and Throughput
Yu [15]	LARAC	Two mixed metrics: Jitter and Packet loss	Jitter	N/D	PSNR
Xu [16]	Random Discretization Algorithm	Two mixed metrics: Packet loss and Delay	Delay	N/D	Packet Loss, Jitter and Throughput
Rego [21]	Dijkstra	Three mixed metrics: Packet loss, Bandwidth and Delay	AI Dynamical Cost Equation to Specify the Constraint between: Loss, Bandwidth and Delay	N/D	Bandwidth Utilization, Packet Loss, Jitter and Delay
Our proposal	TwoLLWPs	Multiple Individual metrics: Available Bandwidth and Packet loss rate	Multiple individual Metrics: Bandwidth and Packet loss rate	SD,HD Ready, HD	SSIM,VMAF and Throughput

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed a flow-based routing strategy for video streaming. The primary goal of the routing strategy emphasized meeting two routing constraints, packet loss, and bandwidth. It efficiently determines feasible paths for each video flow according to the current state of the whole network, like in a smart city. We took advantage of SDN to implement our state-dependent routing scheme by benefiting from the controller global view of the network. It can be used for video surveillance and video delivery in smart cities and smart urban communities. Simulation results have shown that the solution mechanism offers higher video QoE. Moreover, the results suggest that HD videos are impacted more than SD video when there is more packet loss, because it obtains more frame artifacts and color distortions.

This study focuses on investigating packet loss and bandwidth metrics associated with HD and SD. Although the proposed routing algorithm depends on thresholds defined by the administrator, therefore it is possible to implement any other QoS requirements related to video streaming, such as the 4K videos resolution, if the policy defines their loss and bandwidth metrics.

More work is needed to support video QoS-based routing, such as QoE policy enforcement and dynamic resource sharing to provide a method for requesting and reserving network resources for the selected routing paths. Further, we will deploy and investigate the proposed algorithm across inter-domains.

VII. ACKNOWLEDGMENT:

Researchers Supporting Project number (RSP-2021/295), King Saud University, Riyadh, Saudi Arabia.

VIII. REFERENCES

- [1] Seufert, M., Egger, S., Slanina, M., Zinner, T., Hoßfeld, T. and Tran-Gia, P., 2015. A survey on quality of experience of http adaptive streaming. *IEEE Communications Surveys & Tutorials*, vol. 17, no 1, pp.469-92
- [2] McCabe, J.D., 2010. *Network analysis, architecture, and design*. Morgan Kaufmann. Elsevier.
- [3] Costa, L.K. and Duarte, O.B., 1999, June. A scalable QoS-based routing mechanism for supporting multimedia applications. In *Proceedings IEEE International Conference on Multimedia Computing and Systems - ICMCS'99*, vol. 2, pp. 346–351.
- [4] Crawley, E., Nair, R., Rajagopalan, B. and Sandick, H., August 3rd, 1998. A Framework for QoS-based Routing in the Internet. RFC 2386, www.rfc-archie.org.
- [5] Wang, Z. and Crowcroft, J., Sept, 1996. Quality-of-service routing for supporting multimedia applications. *IEEE Journal on selected areas in communications*, Vol. 14(7), pp.1228-1234.
- [6] Costa, L.H.M.K., Fdida, S. and Duarte, O.C.M.B., 2001, June. A scalable algorithm for link-state QoS-based routing with three metrics. In *ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No. 01CH37240) (Vol. 8, pp. 2603-2607)*. IEEE.
- [7] Braden, R., Clark, D. and Shenker, S., June, 1994. RFC1633: Integrated services in the Internet architecture: an overview. Request for Comments (RFC) 1633, www.rfc-archie.org, IETF.
- [8] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z. and Weiss, W., December, 1998. An architecture for differentiated services. RFC 2475, www.rfc-archie.org.
- [9] Rosen, E. and Rekhter, Y., March, 1999. RFC2547: BGP/MPLS VPNs. RFC 2547, www.rfc-archie.org.
- [10] Civanlar, S., Parlakisik, M., Tekalp, A.M., Gorkemli, B., Kaytaz, B. and Onem, E., 2010, December. A qos-enabled openflow environment for scalable video streaming, *IEEE Globecom Workshops on Network of the Future (FutureNet-III)* (pp. 351-356), Miami.. IEEE.
- [11] Egilmez, H.E., Dane, S.T., Bagci, K.T. and Tekalp, A.M., 2012, December. OpenQoS: An OpenFlow controller design for multimedia delivery with end-to-end Quality of Service over Software-Defined Networks. In *Proceedings of the 2012 Asia Pacific signal and information processing association annual summit and conference (APSIPA ASC)* (pp. 1-8). IEEE.
- [12] Egilmez, H.E., Gorkemli, B., Tekalp, A.M. and Civanlar, S., 2011, September. Scalable video streaming over OpenFlow networks: An optimization framework for QoS routing. In *2011 18th IEEE International Conference on Image Processing* (pp. 2241-2244), Brussels, Belgium. IEEE.
- [13] Egilmez, H.E., Civanlar, S. and Tekalp, A.M., May 2013. An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks. *IEEE Transactions on Multimedia*, vol. 15 , no. 3 , pp.710-715.
- [14] Yan, J., Zhang, H., Shuai, Q., Liu, B. and Guo, X., 2015. HiQoS: An SDN-based multipath QoS solution. *China Communications*, vol. 12, no. (5), pp.123-133.
- [15] Yu, T.F., Wang, K. and Hsu, Y.H., 2015, January. Adaptive routing for video streaming with QoS support over SDN networks. In *2015 International Conference on Information Networking (ICOIN)* (pp. 318-323), Cambodia. doi:10.1109/ICOIN.2015.7057904. IEEE.
- [16] Xu, C., Chen, B. and Qian, H., 2015. Quality of Service Guaranteed Resource Management Dynamically in Software Defined Network. *Journal of Communications*, Vol. 10 . No. 11, pp. 843–850. doi:10.12720/jcm.10.11.843-850.
- [17] Sendra, S., Rego, A., Lloret, J., Jimenez, J.M. and Romero, O., 2017, May. Including artificial intelligence in a routing protocol using software defined networks. In *2017 IEEE International Conference on Communications Workshops (ICC Workshops)* (pp. 670-674), Paris, France. IEEE.
- [18] Guo, Y., Wang, W., Zhang, H., Guo, W., Wang, Z., Tian, Y., Yin, X. and Wu, J., 2021. Traffic Engineering in hybrid Software Defined Network via Reinforcement Learning. *Journal of Network and Computer Applications*, Vol. 189, p.103116.
- [19] Liu, W.X., Cai, J., Chen, Q.C. and Wang, Y., March 2021. DRL-R: Deep reinforcement learning approach for intelligent routing in software-defined data-center networks. *Journal of Network and Computer Applications*, Vol. 177, p.102865.
- [20] Rego, A., Sendra, S., Jimenez, J.M. and Lloret, J., 2017, May. OSPF routing protocol performance in Software Defined Networks. In *2017 Fourth International Conference on Software Defined Systems (SDS)* (pp. 131-136), Valencia, Spain. IEEE.
- [21] Rego, A., Sendra, S., Jimenez, J.M. and Lloret, J., September 2019. Dynamic metric OSPF-based routing protocol for software defined networks. *Cluster Computing*, , Vol. 22, No. 3, pp.705-720.
- [22] Go, S.J.Y., Festin, C.A.M. and Tan, W.M., 2019. An SDN-based framework for improving the performance of underprovisioned IP Video Surveillance networks. *Journal of Network and Computer Applications*, Vol. 132, pp.49-74.

- [23] Yamansavascular, B., Baktir, A.C., Ozgovde, A. and Ersoy, C., November 2020. Fault tolerance in SDN data plane considering network and application based metrics. *Journal of Network and Computer Applications*, Vol. 170, p.102780.
- [24] ITU, I., 2008. 1080, Quality of experience requirements for IPTV services. <https://www.itu.int/rec/T-REC-G.1080-200812-I>
- [25] Vega, M.T., Perra, C. and Liotta, A., 2018. Resilience of video streaming services to network impairments. *IEEE Transactions on Broadcasting*, Vol. 64, No 2, pp.220-234.
- [26] Yens K Shortest Path Algorithm, 1970, Yen, Jin Y. (1970). "An algorithm for finding shortest routes from all source nodes to a given destination in general networks". *Quarterly of Applied Mathematics*. Vol. 27, No. 4: 526530. MR 0253822.
- [27] Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. *Numerische mathematik*, Vol. 1, No. 1, pp.269-271.
- [28] Mohanta, K. and Poddar, B., 2012. Comprehensive study on computational methods for k-shortest paths problem. *International Journal of Computer Applications*, . Vol. 40, No. 14, pp.22-26.
- [29] Bouillet, E., Ellinas, G., Labourdette, J.F. and Ramamurthy, R., 2007. *Path routing in mesh optical networks*. Hoboken, NJ: John Wiley & Sons.
- [30] SmartDraw. [Online]. Available at: <https://www.smartdraw.com/about/> (Accessed: 15 June 2020).
- [31] Rec, I.T.U.T., 2008. P. 10: Vocabulary for performance and quality of service, Amendment 2: New definitions for inclusion in Recommendation ITU-T P. 10/G. 100. Int. Telecomm. Union, Geneva.
- [32] Streijl, R.C., Winkler, S. and Hands, D.S., 2016. Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, Vol. 22, No. 2, pp.213-227.
- [33] Wang, Z., Lu, L. and Bovik, A.C., 2004. Video quality assessment based on structural distortion measurement. *Signal processing: Image communication*, Vol. 19, No. 2, pp.121-132.
- [34] Li, Z., Aaron, A., Katsavounidis, I., Moorthy, A. and Manohara, M., 2016. Toward a practical perceptual video quality metric. *The Netflix Tech Blog*, 6(2). [Online]. Available at: <http://techblog.netflix.com/2016/06/towardpractical-perceptual-video.html>.
- [35] Khan, A., Sun, L. and Ifeachor, E., 2008, September. An ANFIS-based hybrid video quality prediction model for video streaming over wireless networks. In *2008 The Second International Conference on Next Generation Mobile Applications, Services, and Technologies* (pp. 357-362), Cardiff, UK. IEEE.
- [36] Zinner, T., Hohlfeld, O., Abboud, O. and Hoßfeld, T., 2010, June. Impact of frame rate and resolution on objective QoE metrics. In *2010 second international workshop on quality of multimedia experience (QoMEX)* (pp. 29-34), Trondheim, Norway. IEEE.
- [37] Li, Z., Bampis, C., Novak, J., Aaron, A., Swanson, K., Moorthy, A. and Cock, J.D., 2018. VMAF: The journey continues. *Netflix Technology Blog*, 25. [Online] Available at: <https://netflixtechblog.com/vmaf-the-journey-continues-44b51ee9ed12>.
- [38] Sintel Trailer. [Online]. Available at: <https://durian.blender.org/> (Accessed: 6 February 2021).
- [39] Caminandes [Online]. Available at: <http://www.caminandes.com/> (Accessed: 19 February 2021).
- [40] Huynh-Thu, Q. and Ghanbari, M., 2012. The accuracy of PSNR in predicting video quality for different video scenes and frame rates. *Telecommunication Systems*, Vol. 49, No. 1, pp.35.