IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 February 2024, accepted 15 March 2024, date of publication 17 April 2024, date of current version 29 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3390577

RESEARCH ARTICLE

QoE-Oriented Routing Mixing Application KPIs and Link Metrics Through Machine Learning

LEI WANG, XIPING WU, (Senior Member, IEEE), AND DECLAN T. DELANEY
School of Electrical and Electronic Engineering, University College Dublin (UCD), Dublin, Ireland
Corresponding author: Declan T. Delaney (declan.delaney@ucd.ie)

ABSTRACT As the complexity of network end devices and applications grows, network managers face increasing difficulty in meeting specific end user requirements, leading to reduced user experience and inefficient resource management. This paper introduces a Quality of Experience (QoE)-oriented routing strategy to enhance user experience by selecting routing paths based on application-specific QoE. Application key performance indicators (KPIs) and dynamic link metrics are utilized to represent real-time QoE and network state. This data builds QoE models for various applications such as video streaming, VoIP, and web map, using four learning methods. The trained models are implemented in a software-defined networking (SDN) controller for optimal QoE routing. Evaluations using the Mininet network simulator reveal that the proposed QoE routing strategy can select the best path 78.4% of the time which is almost 20% more than the top-performing state-of-the-art. This results in measurably higher application performance, proving the efficiency of the proposed approach in improving the application's QoE.

INDEX TERMS Link metrics, machine learning, quality of experience (QoE), routing, software-defined networking (SDN).

I. INTRODUCTION

A. BACKGROUND

Recent substantial technological advancements have led to a significant expansion in network services and applications. This proliferation has given rise to a flexible and diverse range of services, technologies, and connected devices, subsequently increasing the complexity of network management. Moreover, emerging applications, such as self-driving vehicles, holographic meetings, and virtual reality gaming, demand higher network capacities to ensure optimal functionality. While these advanced network technologies have unlocked a multitude of new application services, the sheer scale and demand of these applications have correspondingly escalated the requirements on network capacity. Consequently, the marked increase in both network complexity and capacity requirements has introduced severe challenges to network resource management, especially in scenarios where resources are constrained.

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

Traditional routing strategies, achieved through network design [1], necessitate expert knowledge to determine efficient paths across the network. Dynamic routing, using a singular distributed metric, often results in either over or under-provisioning of resources, failing to meet the nuanced demands of application-specific resource allocation. Thus, routing with a singular metric would often fail in specialized application scenarios. For example, in a network serving both delay-sensitive and bandwidth-sensitive applications, one singular routing metric cannot cater to both needs, presenting a ***problem of specificity***. In addition, identifying the perfect metric for each application adds another layer of complexity, known as the ***problem of optimisation***. A tailored routing metric is required to enhance application performance rather than network efficiency, aligning with the *real* demands of users. The routing strategy must be specific, with a unique metric for each application, and optimized, with each metric carefully designed for the corresponding application.

Although QoS (Quality of Service) routing methods have been popular, they fall short of improving the user experience, especially if service requirements are sensitive to the application's end goals. QoE (Quality of Experience) routing, which harnesses advancements in Machine Learning (ML) and leverages available network data, delves deeper into understanding the real end-user experience, further defining the network's role in shaping applications. As ML technologies mature, they foster more intelligent and adaptive network management systems. By harnessing this network data, we can further automate and refine network management. Through these ML-driven QoE routing strategies, network operators are better positioned to understand and enhance user satisfaction, choosing paths that maximize application performance within the current network landscape.

QoE routing appears promising. If QoE could be measured within the network, network operators would gain better insights into how network performance affects overall customer satisfaction. By exploiting this measurement for routing decisions, network operators could implement QoE-based routing using specific application QoE routing metrics. A QoE routing strategy selects a path that optimizes QoE metric, with unique QoE metrics to evaluate different applications or groups. Strategies for QoE routing have been developed by quantifying the relationship between network state metrics and target QoE metrics, applying heuristic, expert opinion, or ML approaches to create specific QoE routing models [2], [3], [4]. A QoE model integrates **user perception** and **network state**, developing a tailored relationship for each application [5]. This allows network managers to select paths that enhance the performance of the application based on the current network state.

B. RELATED WORK

QoS routing is one of the most widely used routing strategies for improving network service. IETF has described QoS-based routing issues and requirements [6]. Earlier routing algorithms, such as Bellman Ford's [7] and Dijkstra's [8], are used to find the shortest path for packet transmission. Later, many QoS routing methods are achieved by involving one set of constrained network metrics while optimizing the others [9], [10], [11]. For example, the Widest-Shortest path (WSP) algorithm [12], [13] determines the shortest paths first and then selects the one with the most width, whereas the Shortest-Widest path (SWP) algorithm [12], [14] considers the widest bandwidth paths first and then selects the one with the least hop count. Among these proposed routing algorithms based on composite network metrics, the bandwidth-delay-constrained routing problem is a focus in the literatures [15] and [16]. These QoS routing strategies, however, don't consider the user's experience with the application. They don't test how well the application works. Therefore, there's a need to create better routing methods focused on user experience, called QoE-driven routing.

The QoE strategy paradigm was introduced to directly enhance application performance. It was tested and developed for application-specific gains, and has been applied to improve 5G and beyond 5G, as mentioned in [17] and [18].

For example, Nightingale et al. [18] have proposed a 5G-QoE framework to address the QoE modeling for UHD video flows in 5G networks. Lemeshko et al. [3] developed and researched a model of adaptive routing with the provision of QoE. In their work, the application's QoE is calculated as an R-factor from the E-model. Nam et al. [19] also used a QoE-aware routing strategy for video streaming. These work have all aim to improve the vatious application's performance applying the QoE strategy.

While several QoE routing strategies exist, not all use *real* QoE metrics that genuinely reflect the user's perspective. For instance, Nam et al. [19] employ video streaming startup latency, buffering rate, and playout buffer status as QoE indicators. However, these metrics may not directly correlate to human perceptual experiences. Similarly, Barakabitze et al. [20] focus on minimizing video quality switches and startup delays to enhance the end-users' QoE, yet the direct relevance of these metrics to a user's real experience remains unclear. A kind of QoE metrics that can represent the real experience of users needs to be measured.

Aiming to capture *real* user experiences, some researchers have turned to more direct QoE metrics. When deciding on these metrics, there's a distinction to be made between subjective and objective QoE indicators [21]. Liu et al. [2] use a human-centric scoring system for virtual reality video QoE. These subjective QoE metrics, representing the user experience rely on methods such as user testing and surveys. But these methods can be logistically complex and challenging to quantify in real-time. To address these challenges, machine-measurable QoE metrics have gained traction. Hu et al. [22] employed ML to associate network metrics with PESQ, a VoIP KPI. They also explored the influence of QoS attributes on PESQ. Similarly, Jie et al. [23] utilized PSNR as a video quality KPI that signifies user QoE. These application KPIs metrics are machine-measurable and can be mapped to reflect the user's perspective.

Efficiently measuring network state metrics is crucial for QoE routing strategy as well. There are principally two methods for this: End-to-End (E2E) metric and link metric. E2E metrics evaluate selected pathways between two points, let's say A and B. A significant benefit of this approach is that testing is initiated only when a specific path is requested, ensuring that the assessed path precisely mirrors the anticipated available resources. E2E metrics have gained traction in QoS routing [24], [25] as well as QoE routing [2], [3], [22] to represent the network state. E2E test, faces challenges as well. One major concern is the costly testing procedures involved [26]. In addition, if a call must wait for the entire E2E test to conclude before being routed, this can lead to longer path setup times, thereby causing delays. Such delays can be detrimental, especially when considering applications like voice calls. The International Telecommunication Union (ITU-T) G.114 states that for optimal voice quality, the one-way delay should not exceed 150 ms [27]. Hence, meeting such stringent delay requirements becomes challenging with E2E tests. Another limitation with E2E testing is that once a path is designated, it remains static unless retested. Therefore, applying E2E metrics in routing strategy still faces challenges such as expense, time-consuming processes, stale data, etc.

In contrast, link metric collection assesses individual links or shares switch data, indicating the state of links to the network manager. It quickly provides path information between points *A* and *B*, offsetting E2E collection limitations. In the field of network design, link metrics are gaining prominence. Munaretto et al. [28] implemented a QoS-centric routing method within the Optimized Link State Routing (OLSR) protocol using link state metrics as an indicator for selecting the best routing path. Similarly, Thorpe et al. [29] utilized iMOS, an intermediate MoS link metric, to enhance VoIP QoS monitoring at intermediate nodes in an OpenFlow SDN. This method highlights the potential of SDN-derived link metrics in offering efficient, real-time VoIP monitoring. It's evident from various researches that SDNs are frequently leveraged to collect link metrics [4], [30], [31].

While link metrics have been incorporated into QoE routing in previous research [19], [20], these studies did not employ a learning approach to establish a connection between the link metrics and genuine QoE metrics. Instead, they continued using non-authentic QoE metrics. A comparison between E2E and link metrics for modeling QoE in VoIP and video applications was conducted in a previous study [4], [30]. The findings from this research suggest that link metrics offer comparable prediction accuracy to E2E metrics models in QoE modeling. Consequently, link metrics present an opportunity to serve as a viable alternative to E2E metrics. This research allows individuals to choose between these two network metrics based on their specific requirements. Given the shortage of E2E metrics, the objective of this paper is to assess the efficacy of using link metrics in QoE strategy.

To model QoE, ML methodologies can be employed to establish the relationship between QoE and the network state. ML-based approaches have a learning nature, and predicted QoE can always be used as feedback in ML processing [32]. So far, ML has been extensively applied in modeling QoE [33].

For instance, Hu et al. [22] tapped into ML to correlate QoS metrics with PESQ, a prominent VoIP QoE KPI, and to unearth the impact of various QoS attributes on PESQ. In a different vein, Abar et al. [34] proposed a QoE prediction model grounded in comprehensive parametric and application metrics within SDN architectures. This prediction model, derived from four unique learning algorithms, is geared towards evaluating the QoE for video. Further, the research by Wang et al. [4], [30] used ML to create QoE models for two applications and compared the efficacy of various ML algorithms in QoE modeling. This work demonstrates the efficacy of using ML techniques to analyse the relationship between application QoE and network metrics.

A comparison of QoE routing strategies is detailed in Table 1. This table emphasizes the QoE metrics and network

TABLE 1. QoE routing strategies comparison.

Ref.	Year	Metrics Modeling: Real QoE	Metrics Modeling: Network State	ML	SDN	App Type
[19]	2014		Link		✓	Video
[20]	2018		Link		✓	Video
[3]	2020	✓	E2E		✓	VoIP
[22]	2020	✓	E2E	✓		VoIP
[2]	2020	✓	E2E	✓	✓	Virtual Reality
[23]	2022	✓	Not Specified	✓		Video
This Work		✓	Link	✓	✓	VoIP, Video, Web map

state metrics employed in QoE routing, categorizing them based on two criteria: the utilization of *real* QoE metrics and whether E2E metrics or link metrics are used to depict the network state. Furthermore, the table shows the growing adoption of SDN and ML techniques in QoE routing as new technologies and tools emerge. Given the challenges of defining QoE metrics and the known limitations of E2E metrics, as shown in the last row of the table, this paper needs to explore the application of tailored QoE metrics that can directly represent the user's perspective and be measured by machines. In addition, the focus will be on using efficient network metrics, like link metrics. Building on this, the paper plans to leverage the advantages of SDN and ML to conduct experiments and model these metrics.

C. CHALLENGES AND MOTIVATION

1) TAILORED QOE METRICS ARE DIFFICULT TO DEFINE

QoE routing strategies aim to enhance the performance of individual applications by incorporating metrics that reflect the perceived performance. However, a significant limitation of earlier QoE routing approaches is their reliance on metrics that don't directly correspond to the user's defined experience. While numerous studies on QoE use metrics, they often lack a clear link between these metrics and user perception indicators, such as the Mean Opinion Score (MOS). Although QoE metrics with a direct tie to MOS are available and frequently used, measuring user experience currently relies on methods like testing and polling, which are logistically challenging and difficult to quantify in real-time. The challenge of pinpointing appropriate QoE metrics has made it tough to expand the use of such methods in specialized QoE routing plans. Thus, the task remains: how to define scalable, machine-readable QoE metrics tied to MOS benchmarks for effective QoE routing.

Recent studies have identified a correlation between application QoE and real-time KPIs. Although these subjective QoE metrics are produced by end-users and aren't directly accessible to network managers, KPIs metrics can be used as machine-readable indicators of the QoE. It's also important to note that when applications measure their KPIs metrics, they can send these metrics to the controller in real-time using the SDN Northbound interface, thus making the application KPIs accessible. Consequently, KPIs can function as a proxy measurement for application QoE. The QoE and networking community [35] has defined cost metrics for dynamic applications used in network management and QoE measurement. This paper will evaluate the performance of multiple applications across the network and compute their respective QoE KPI metrics.

2) EFFICIENT NETWORK STATE METRICS

QoE routing strategies necessitate current network state information to optimize paths within the network. While E2E metrics are frequently used in network design, they come with limitations. Drawbacks include extended time needed for path setup during testing, costly testing procedures, static path designation unless retested, and non-availability until the entire test is completed. Therefore, a more efficient network metric is sought.

Link metric collection periodically evaluates individual links or shares switch counter information to reflect the state of connected links with the network manager. This method benefits from instant availability of path information, mitigating the disadvantages of E2E metric collection. A shortage to note is that composite link metrics might not wholly represent all network resources.

Compared to E2E metrics, network link metrics offer more up-to-date data, faster testing, reduced overheads, and instant data availability. Recognizing these advantages, this paper leans towards link metrics. For gathering this data, SDN tools are used. The controller ensures consistent data collection and updating, sourcing information from OpenFlow messages. Therefore, this paper finds it essential and valuable to examine the efficacy of using link metrics in constructing QoE models, subsequently influencing QoE routing decisions.

3) QOE MODELING USING APPLICATION KPIS AND LINK METRICS

A challenge still remains in effectively modeling QoE using the existing, available metrics. For example, E-model uses computational models to combine factors like network delay, packet loss, and noise to predict VoIP quality. It calculates the R-factor from these parameters, which can then be mapped to the MOS, indicating user QoE. The "R-factor" in E-model calculations is a quality rating value indicating transmission quality, convertible into metrics like the MOS for assessing user experience. The E-model does not collect subjective metrics within the network, leading to a lack of real-time feedback and rendering the model unsuitable for Reinforcement Learning (RL) in future.

ML-based approaches are designed to update when new information is available allowing measured QoE/KPIs to be used as feedback in ML processing if available. ML can establish the relationship between application QoE (KPI metrics) and network state (link metrics), thereby building a QoE model. This model predicts the application KPI. In future work, this prediction can be combined with feedback from end users. Such collaboration could be employed in RL to enhance model efficacy. Using ML can meet the requirements of modeling QoE using KPIs and link metrics, and is also well-suited for expansion into RL. Therefore, employing ML in QoE modeling is essential for this study.

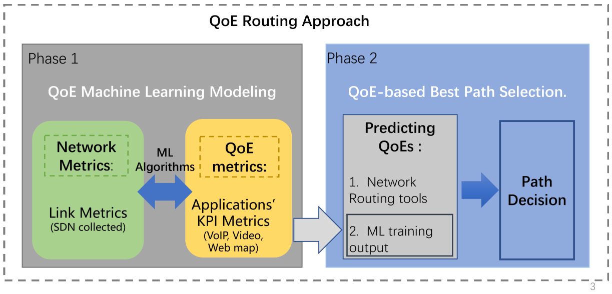


FIGURE 1. **QoE routing approach.**

D. APPROACH AND CONTRIBUTIONS

The paper defines a QoE metric using application specific, machine readable, application layer KPIs. The network state is updated in real-time by the network manager using link metrics. Best routing paths between A to B are chosen by highest predicted KPIs using composite link metrics between the two. The proposed approach to achieve QoE routing is illustrated in two distinct phases in Fig. 1: 1) QoE ML Modeling, and 2) QoE-based Optimal Path Selection. First, ML is applied to train the QoE model using application KPIs and link metrics. Next, this model predicts the KPI for each path, selecting the one with the highest KPI. The goals of the approach are then to identify and define suitable KPIs for a given application that have a tangible relationship with QoE, and develop a model to predict the KPI score for a path defined by composite link metrics.

The key contributions of this paper are as follows:

- Developing a novel QoE model that compounds application KPIs with link metrics using ML techniques. In this approach, application KPIs metrics are defined and measured as the QoE, and the link metrics are sourced from an SDN testbed. The development of this QoE model is facilitated using four ML algorithms. Subsequently, the efficacy of this novel model is assessed across three key Internet applications: VoIP, video streaming, and Web map.
- Designing a two-stage QoE Routing Algorithm. In the first stage, network with complex topology is broken down into distinct paths using conventional routing methods, such as the K shortest simple path. During the second stage, the QoE model is utilized to estimate the application KPI for a subset of selected paths, ultimately choosing the route that delivers the highest QoE.
- Developing an SDN testbed using Mininet [36] to assess the proposed QoE routing within an OpenFlow environment. This testbed operates on an open-source codec

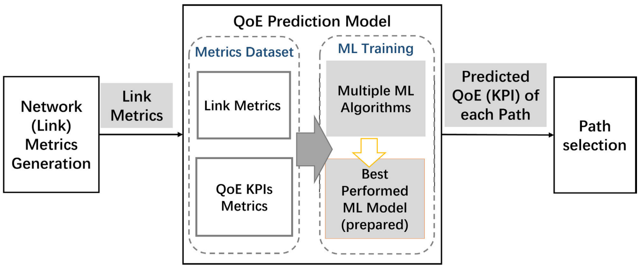


FIGURE 2. Fundamental approach of QoE routing system model.

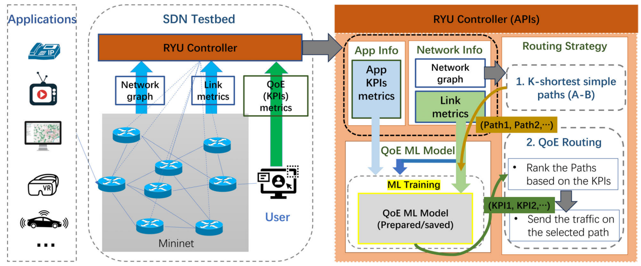


FIGURE 3. QoE routing framework.

available on GitHub [37]. The codec comprises key functionalities including running Internet application tests, gathering link metrics, integrating QoE models, and executing the QoE routing algorithm.

- Experimental results:
  - The results show that the proposed QoE routing algorithm can select the best path with a probability of 78.41% which is nearly 20% higher than the top-performing state-of-the-art.
  - Feature analysis indicates that different applications respond uniquely to network QoS levels. This underscores that a one-size-fits-all QoS strategy cannot cater to diverse applications, highlighting the need for tailored QoE models for specific application.
  - The efficacy of compound metrics, which combine application KPIs and link metrics, is thoroughly examined using four ML algorithms. Results reveal that Random Forest (RF) achieves the best evaluation value among them.

II. PROPOSED FRAMEWORK

Fig. 2 illustrates the fundamental approaches of the QoE routing system model and explains how it operates. In the beginning, link metrics such as delay, packet loss, and bandwidth need to be generated. If the network manager can gather these network metrics for a path, they can be used as input in the QoE prediction model. The prediction model, typically developed using ML, determines the relationship between a QoE (KPI) metric and measurable network (Link) metrics. This allows the network manager to evaluate any given path and assign it a QoE metric, in this case, the application KPI. This relationship (model) must be known to the manager before path selection. Following that, a set of predicted QoEs (application KPIs) is collected, and the routing is determined by selecting the path(s) with the highest QoE(s). This section will investigate the QoE routing system and the methodology: a) The QoE routing implementation framework; b) Application QoEs; c) Network State Metrics; d) ML algorithms used in QoE modeling.

A. QOE ROUTING IMPLEMENTATION FRAMEWORK

Fig. 3 presents the framework of QoE routing implementation. This framework illustrates the QoE routing testbed and the workflow. In the testbed, Various application servers like VoIP, video, and VR gaming, all connected to an SDN network. The SDN network is emulated using Mininet, simulating a realistic network environment with SDN switches, controllers, and multiple path links. The Ryu functions as the SDN controller. A user/client connected as a host receives network packets from the other end of the network.

Before the real-time QoE routing process, there's an offline ML modeling phase using data collected from experiments. As soon as the network starts, the Ryu controller begins collecting link metrics from the SDN testbed. Once all the packets are transmitted, the user's end computes the QoE (KPIs) metrics tailored for the various applications. Both sets of metrics are compiled to form the training dataset for buidling the QoE prediction models. Consequently, the QoE prediction models are prepared and ready for use.

Fig. 3 also presents an run-time implementation of QoE routing that aims to present the best path available to the network manager. In this phase, the network manager collects link metrics from SDN switches using Ryu. These metrics are related to the Network Info block, which stores the network graph as well as the network metrics associated with the links in the graph, and is updated periodically. When a path from A (application server) to B (user/client) is requested, the network graph is consulted. In this context, a two-stage routing algorithm is employed

1) A set of K shortest simple paths from A to B is derived from the graph. The K shortest simple path problem generates all simple paths from the source to the target in the graph, starting with the shortest [38]. Notably, a simple path is one that does not contain any repeated nodes. Consequently, a set of paths (path1, path2, …) are gathered from the network graph using the K shortest path strategy;
2) The previously trained QoE model utilizes the link metrics associated with each path from the aforementioned set. This utilization facilitates the prediction of QoE for each individual path. It is expected that the application type will be detected through application identification, which is outside the scope of this paper. The QoE ML model provides a predicted QoE (KPI) for each path, generating a list of these KPIs (KPI1, KPI2, …) that is presented to the network manager. These KPIs are then ranked to determine the best-performing

path. Subsequently, the manager can guide the traffic flow onto the path with the highest QoE, thereby implementing QoE routing.

B. APPLICATIONS QOES

QoE metrics can be classified into two types based on the measurement mechanisms used: objective and subjective metrics [39]. Subjective QoE is qualified and collected from the end user, reflecting the user's perspective and personal concept of "good quality." Due to its inherently subjective nature, its assessment requires a controlled environment, making it more dependent on context and often resulting in high costs and extensive evaluation time. For example, subjective tests may need numerous volunteers to score a service, and scores can vary depending on factors such as age, background noise, health status, etc. As subjective QoE metrics are challenging to measure, researchers are modeling QoE with service KPIs, bridging the gaps between subjective and objective metrics. Objective quality metrics, such as application KPIs, are more scalable and machine-measurable than subjective metrics.

In this paper, objective QoE, application KPIs modeling is applied for three applications:

1) **VoIP KPI Measurement:** PESQ: Perceptual Evaluation of Speech Quality (PESQ) is more practical, comparing the received and original audio to create an objective indicator of quality [40]. It is standardized as Recommendation ITU-T P.862 and ranges from -0.5 to 4.5, with higher scores indicating better quality.
2) **Video KPI Measurement:** Traditional methods like Peak Signal-to-Noise Ratio (PSNR) [41], Video Multi-Method Assessment Fusion (VMAF) [42], Structural Similarity (SSIM) [43] often fail in lossy networks.
   APSNR: To tackle the inaccuracy caused by frame mismatch in lossy networks, an optimized measurement algorithm of video quality, Aligned-PSNR (APSNR), is implemented [4], [44]. APSNR aligns frames between reference and received videos, making it effective for measuring quality in frame-loss transmissions.
3) **Web KPI Measurement:** Web QoE refers to the quality of experience of web services accessed via a web browser [45]. Map PLT: In the context of Web map applications, Map Page Load Time (PLT) is used as the QoE KPI metric. It measures the time from when the user requests the map service until the map tiles are completely loaded. This objective Application-level Web QoE metric, like VoIP and video QoE, can also be mapped to the MOS value [46].

C. NETWORK STATE METRICS

1) E2E AND LINK METRICS

E2E metrics represent a type of network quality metrics that have been widely utilized due to their close alignment with the actual network QoS, providing reliable and accurate network state information. As the name suggests, E2E metrics are collected along a single path, tested between the source and destination, as depicted in Fig. 4. These metrics usually include latency, bandwidth, jitter, and packet loss. Link metrics, on the other hand, utilize a composite of network metrics for each link in the path, as shown in the figure. Contrary to E2E metrics, link metrics offer a more dynamic approach to data collection and can provide real-time network information. This is in part due to the drawbacks of E2E metrics, which can be time-consuming, expensive, and result in stale information, as well as being unavailable until the entire test is completed. With link metrics, the SDN controller gathers data from each link and frequently updates it. This method is not only faster but also avoids occupying the main transmission route.

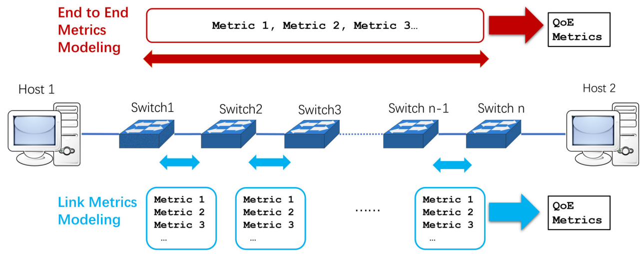


FIGURE 4. E2E and link metrics.

Previous work [4], [5], [30] presents arguments in favor of using link metrics in this context. They emphasize the trade-off between efficiency and prediction performance when choosing between E2E and link metrics. While link metrics modeling benefits from efficient metric collection, E2E metrics modeling may provide slightly better prediction performance. Given these considerations and the inherent advantage of real-time representation of network state, link metrics have been chosen in this paper as the preferred method for network data collection.

2) LINK METRICS COLLECTION

The collection of link metrics is achieved by using an SDN. Building on previous research [4], [30], link metrics are calculated using information extracted from OpenFlow messages. When connected to an SDN controller, the OpenFlow switches communicate messages such as Packet_in, Packet_out, STATISTICS_REQUEST, and STATISTICS_REPLY. These messages provide information that can be analyzed and used to calculate link metrics, including link bandwidth, link delay, and link packet loss. For example, link metrics are defined as the statistics on the link between two switches, which can be named S1 and S2.

Link bandwidth is calculated as:

The equation(s) to calculate Link bandwidth

$$Bandwidth_{link} = BW1 - \frac{tx\_packets - rx\_packets}{duration\_time} \quad (1)$$

where, BW1 is the default bandwidth of the link, as given in the assumption. The terms rx_packets and tx_packets represent the amounts of received and transmitted packets, respectively, and duration_time refers to the duration that the packets pass through a switch.

Link delay is calculated as:

The equation(s) to calculate Link delay

$$T1 = time2 - time1, T2 = time4 - time3 \quad (2a)$$

$$T_a = RTT_a, T_b = RTT_b \quad (2b)$$

$$Delay_{link} = \frac{T1 + T2 - T_a - T_b}{2} \quad (2c)$$

where, the parameters refer to the time recorded when:

time1: the controller sends a Packet_out message to S1

time2: the controller receives the Packet_in message sent by S2

time3: the controller sends a Packet_out message to S2

time4: the controller receives the Packet_in message sent by S1

Link packetloss is calculated by dividing the packets amount difference between two switches by the transmitted packets in (3):

The equation(s) to calculate Link packetloss

$$Packetloss_{link} = \frac{tx\_packets(S1) - rx\_packets(S2)}{tx\_packets(S1)} \quad (3)$$

where, tx_packets(S1) are the transmitted packets from S1 count and rx_packets(S2) are the received packets count at the S2.

SDN monitors and updates the network information frequently by sending OpenFlow messages. Periodic or event-based messages are sent to the controller, providing the information needed to calculate the link metrics. The frequency of these messages or events determines the freshness of the network state. As these link metrics are derived from network layer content and statistics, they are considered to represent the network state in real-time for QoE prediction.

D. ML ALGORITHM USED IN QOE MODELING

Selecting suitable ML tools to model the relationship between network state (metrics) and QoE (metrics) is a significant task. The choice of the appropriate learning method for a project requires an understanding of the ML type, project goal, data set size, required training time, feature/parameter analysis, and availability of labeled data. Factors such as data size and training time are particularly important when considering the algorithm type. With small data sets and limited training time, the project may lean toward less complex or ensemble learning processes, where over-fitting is less likely to occur. In addition, a feature analysis, encompassing aspects like dimensionality and feature type, is critical for determining the appropriate learning method.

The expected output of the model is a predicted QoE value. Since the target feature is available during testing in a testbed or as application feedback in network learning, supervised forecasting algorithms are deemed suitable. Given that the data used for training is continuous and labeled, supervised regression algorithms fit the requirement. The data set's features, such as link bandwidth, link delay, and link packet loss, are of low dimensionality, and the data set size is expected to be in the thousands of data points rather than millions. In this context, Supervised Learning algorithms are considered appropriate.

Various supervised regression algorithms are available:

- Multiple Linear Regression (MLR) models the linear relationship between a quantitative dependent (target) variable and two or more independent (descriptive) variables. It offers simplicity and less complexity, providing a baseline result. Its main drawback is that it assumes a linear relationship among variables, oversimplifying real-world issues, and may thus be unsuitable for practical scenarios.
- Support Vector Regression (SVR) is robust to outliers and typically has higher prediction accuracy than MLR, especially for continuous data. While SVR struggles with large and noisy data sets, this limitation is mitigated in the case as noise can be cleaned via pre-processing.
- RF can handle large data sets and complex, non-linear relationships. Its interoperability and performance with large RF ensembles make RF suitable for training our data set, which consists of approximately 76,000 samples.
- Gradient Boosted Regression (GBR), like RF, uses decision trees and can even outperform RF if properly tuned. It serves as another viable option for comparative training.

To investigate the efficiency of learning the relationship between network state and QoE, these four ML algorithms will be applied in training the QoE prediction models. After evaluating and comparing their performance, the best training algorithm will be selected and used in building the QoE prediction model. This thorough analysis will ensure that the chosen methodology aligns with the specific needs and characteristics of the data and project at hand.

III. PROPOSED ROUTING ALGORITHMS

A. QOE ROUTING ALGORITHM

QoE routing selects the path based on the QoE model's prediction of the QoE (application KPI) for each path. The prediction models for each application and each path have been trained and prepared offline. When the application's traffic arrives at the switch, the controller identifies the application's type based on the classification output from the Application Identification API. This Identification API may include tools such as DPI (Deep Packet Inspection) or other packet identification tools, as well as application classification models. However, it is important to note that these are not within the scope of this paper. The application-specific QoE models are invoked according to this traffic type. Subsequently, using the collected link metrics as input, ML models are applied to predict the real-time QoE. Regular updates are necessary to ensure that the input data remains fresh. Once the QoE for each path has been predicted, a decision is made based on these QoE values. Users can select their preferred path(s) corresponding to specific QoE. Naturally, in most cases, the path with the highest QoE will be chosen.

This QoE routing strategy is executed in two stages. We use pseudo code to illustrate the process of implementing the QoE routing strategy. In Algorithm 1, the inputs are the network topology/graph G, link metrics (link BandWidth BW_{link}, link Delay D_{link}, link PacketLoss PL_{link}), trained prediction Models (*ML_models*) representing ML model groups for multiple applications, and the application type *App_Type*. The output is the *best* routing path, denoted as *Best_Path*(s).

Stage 1: Before selecting the path, complex network topologies, such as mesh, must be broken down into simple paths using traditional routing algorithms. This is the first step in the algorithm. The K shortest simple path policy, provided by the NetworkX *simple_path*() function, can be used to divide the network into several simple paths between A and B stored in *Paths*. The *src* and *dst* represent the source and destination nodes in the network. This process of breaking down paths is also executed by the comparative routing algorithms in Section III-B.

Stage 2: The specific QoE model will be selected and saved as **Smodel** in Step 2, based on the application type *App_Type*. This selected model will be invoked and used to predict the corresponding QoE for each path in Step 3. The link metrics are used as input, and the path QoE is output if the ML models successfully produce a result. This QoE is saved in a QoE list that will be ranked in the final step. Users can select the path(s) based on the QoE values. They may choose one path with the best QoE, or several paths whose QoEs meet or exceed their requirements.

B. COMPARATIVE ROUTING ALGORITHMS

One widely used approach in QoS routing is Bandwidth-Delay constrained routing, which provides a sound methodology for choosing a path based on the combined state of multiple network features. The implementation of Bandwidth-Delay constrained routing requires prior knowledge of the bandwidth constraint, which represents the minimum bandwidth required to transmit the application data. In the case of VoIP, this constraint depends on the type of audio codec. For example, G.711 requires an IP bandwidth of 80-90 Kbps, while an HD VoIP call typically uses 90-100 kbps of bandwidth. Therefore, 100 Kbps of bandwidth must be guaranteed to make a G.711 VoIP call. For video, the constraint value is determined by factors like video size and resolution. In [4], a significant inflection point of 1.25 Mbps bandwidth is observed during the transmission of a Common Intermediate Format (CIF) video. Below this inflection point, the APSNR over the bandwidth range of 0.25 Mbps to 1.25 Mbps rises sharply, whereas afterward, it tends to stabilize. Thus, 1.25 Mbps is the bandwidth constraint for CIF video. In the case of web map loading, a different behavior is observed. There is a steep decrease in loading time before 5 seconds, followed by a gradual slowdown. Consequently,

Algorithm 1 QoE Routing

Input : G, BW_{link}, D_{link}, PL_{link}, *ML_models*, *App_Type*
Output: *Best_Path*

STEP 1: Breaking the network paths
if *src, dst in G* **then**
  Paths = *networkx.all_simple_paths*(G, *src*, *dst*)
STEP 2: Select the ML models based on the application type
Smodel = *ML_models* (*App_Type*)
STEP 3: Predict QoE on each path
for *p in Paths* **do**
  call *Smodel*;
  QoE (p) = **Smodel** ($BW_{link1}, D_{link1}, PL_{link1}, BW_{link2}, D_{link2}, PL_{link2}, \ldots BW_{linkn}, D_{linkn}, PL_{linkn}$)
  if *prediction success* **then**
    add *QoE* (p) to *Paths_QoEs*
STEP 4: Select the path
rank the *Paths_QoEs*;
Best_Path(s) = the satisfying path(s)

0.6 Mbps bandwidth is applied as the bandwidth constraint for web map loading. By understanding and applying these constraints, Bandwidth-Delay constrained routing is able to effectively select paths that align with the specific requirements of various applications, enhancing overall network performance.

The implementation of Bandwidth-Delay constrained routing shares the same Step 1 as QoE routing's, but with different input parameters. In Step 2, the E2E bandwidth (BW_{E2E}) and E2E delay (D_{E2E}) on each path are calculated using the link metrics. Then in Step 3, the *best* path is selected based on BW_{E2E} and D_{E2E}. There are three cases, divided according to the relationship between BW_{E2E} and threshold BW_T.

- Case 1: If all the paths' bandwidths meet the constraints requirement, the *best* path will be selected from the whole path list *Paths*, choosing the one with the minimum delay.;
- Case 2: If only some of the paths' bandwidths meet the requirements, the guaranteed paths are all saved in *k_Paths*. The *best* path will be the one with the minimum delay, selected from this *k_Paths* list;
- Case 3: If none of the paths meet the bandwidth requirement, bandwidth is considered a priority, and the path with the maximum bandwidth is chosen.

These three cases ensure that the selected path meets the requirements for bandwidth and delay according to the specific conditions and constraints of the network.

The mechanism to implement this algorithm can be easily adapted for the Bandwidth-Packetloss constrained routing by substituting the delay parameter with packet loss. In addition

Algorithm 2 Bandwidth Delay Constrained Routing

Input : $G, BW_{link}, D_{link}, PL_{link}$
Output: *Best_Path*

STEP 1: Breaking the network paths
if *src, dst in G* **then**
  $Paths = networkx.all\_simple\_paths(G, src, dst)$
STEP 2: Calculate the BW_{E2E} and D_{E2E}
for *p in Paths* **do**
  $p.BW_{E2E} = \min(BW_{link1}, BW_{link2}, \ldots BW_{linkn})$
  $p.D_{E2E} = \mathrm{sum}(D_{link1}, D_{link2}, \ldots D_{linkn})$
STEP 3: Select the best path
if *all paths'* $BW_{E2E} \geq BW\_T$ **then**
  Best_Path = *Paths* [path with min D_{E2E}])
else if *k paths'* $BW_{E2E} \geq BW\_T$, *and* $k < len(Paths)$ **then**
  for *p in Paths* **do**
    if $p.BW_{E2E} \geq BW\_T$ **then**
      add *p* to *k_Paths*
  Best_Path = *k_Paths* [path with min D_{E2E}]
else
  Best_Path = *Paths* [path with max BW_{E2E}]

to these multi-parameter-based routing approaches, there have been past implementations that rely on single network parameters, such as Bandwidth-only routing, Delay-only routing, and Packet-loss-only routing.

IV. EXPERIMENTATION AND RESULTS

A. TESTBED CONFIGURATION

The experiments will test three applications—VoIP, video, and Web maps—using a configuration process detailed in earlier works [4], [5], and [47]. As shown in Fig. 5, the testbed includes three sections: Application server, SDN network, and application user/client, each on distinct Virtual Machines (VMs). Communication between the server and client utilizes three applications: SIPp, VLC, and Web map based on OpenstreetMap. Link conditions and routes can be adjusted as needed.

VM1 Multi-applications' Server: The multi-applications server is in VM1 (Ubuntu 18.04 X86 64 bits, one CPU, 4.0 GB RAM). Three applications are integrated:

- VoIP Call: SIPp simulates VoIP calls, sending RTP packets. A 33-second English voice speech is sent using the G.711 codec;
- Video Streaming: VLC streams a 20-second clip from "highway" video, with a resolution of 352 * 288 pixels;
- Web Map: Built using OpenStreetMap (OSM), Ireland's map is stored in PostgreSQL, with Apache serving and rendering OSM tiles.

VM2 Multi-applications' User/Client: The user/client is on VM2 (Ubuntu 18.04 X86 64 bits, one CPU, 2.0 GB RAM). SIPp and VLC receive VoIP and video, respectively. For

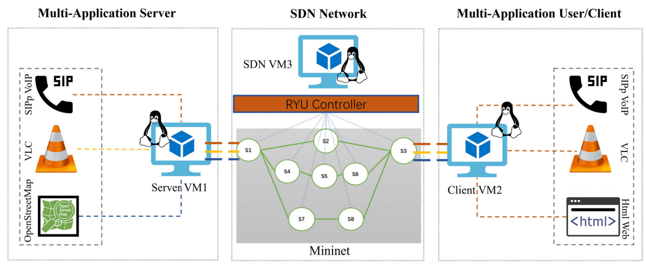


FIGURE 5. Testbed configuration.

Web maps, a NodeJS-developed HTML page displays on the Chromium browser, with a zoom level set to 10 and default latitude, longitude coordinates corresponding to Ireland.

VM3 SDN Network Setup: In VM3 (Ubuntu 18.04 X86 64 bits, one CPU, 2.0 GB RAM), a 3-path network is built using Mininet with Ryu as the controller. Fig. 5 shows 8 OpenFlow switches, forming a series of independent paths connecting the client and server. Path lengths of 2-link, 3-link, and 4-link can extend to n-links. Ryu monitors and collects link metrics data.

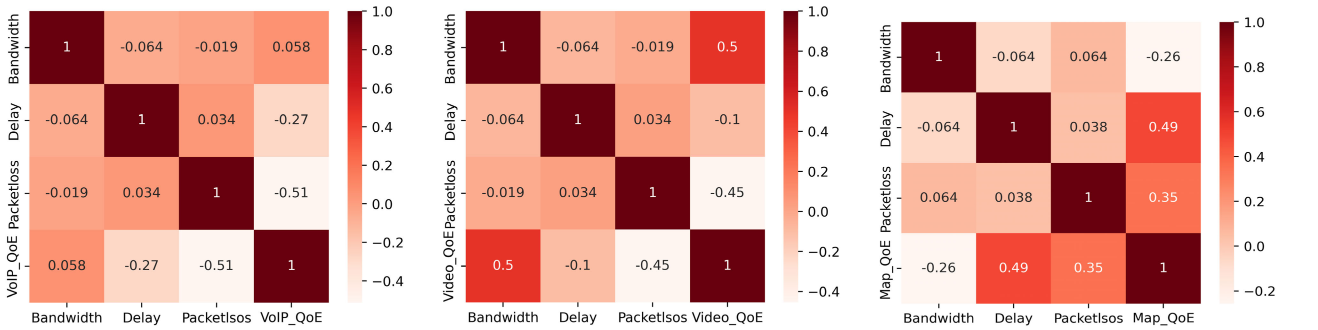
B. FEATURE ANALYSIS

An analysis of the impact of network features such as Bandwidth, Delay, and Packet Loss on the QoE of various applications is conducted using end-to-end (E2E) network QoS metrics. These metrics are derived from link data collected by the Ryu controller. The Pearson Coefficient Correlation (PCC) is employed to gauge the relationship between these features and the QoE for different applications (PESQ, APSNR, Loading Time), as shown in Fig. 6. The PCC, ranging from -1 to 1, reflects the strength and direction of the correlation, with the absolute value indicating the robustness of the relationship.

Figure 6 highlights how packet loss, bandwidth, and delay as network features primarily influence the QoE of VoIP (PESQ), video (APSNR), and Web maps (loading time), respectively.

For VoIP, Fig. 6(a) reveals a strong negative correlation between packet loss and PESQ (-0.51) and a weaker correlation with bandwidth (0.058). This makes sense, given that G.711 VoIP typically requires only about 100 Kbps of bandwidth, which is sufficient in most cases. In contrast, packet loss can severely degrade call quality by causing choppy audio and dropped calls. Delay, often linked to long-distance transmission exacerbates these issues. Consequently, VoIP PESQ is most sensitive to packet loss and delay, less so to bandwidth.

For video streaming, Fig. 6(b) shows significant impact from bandwidth and packet loss on APSNR, with PCC values of 0.5 and -0.45, respectively. High-resolution video requires substantial bandwidth, and packet loss can cause pixelation, content gaps, and playback failure, all lowering APSNR. These factors explain why bandwidth and packet loss are strongly connected to video APSNR.



(a) VoIP PCC (b) Video PCC (c) Web Map PCC

FIGURE 6. Network features pearson coefficient correlation with QoEs of 3 applications.

TABLE 2. P-values of network features.

	VoIP		Video		Web Map	
	PCC	P-value	PCC	P-value	PCC	P-value
Bandwidth	0.058	0.0957 >0.005	0.5	9.206e-52 <0.005	-0.26	9.84e-14 <0.005
Delay	-0.27	2.26e-15 <0.005	-0.1	0.003 <0.005	0.49	2.81e-49 <0.005
Packet -loss	-0.51	7.48e-56 <0.005	-0.45	1.11e-42 <0.005	0.35	5.02e-25 <0.005

For Web maps, Fig. 6(c) shows that delay, bandwidth, and packet loss all affect loading time, though delay has the highest PCC (0.49). Insufficient bandwidth can lead to congestion and slow loading, while packet loss cause delays through TCP retransmission. All three features impact the Web map's QoE, with delay having the most substantial effect.

The significance of the correlation between network features and applications' QoE is evaluated using a hypothesis test for the correlation coefficient. Utilizing a p-value, which represents the level of statistical significance, results are considered significant if the p-value is less than 0.05. As shown in Table 2, most network features for the three applications exhibit p-values smaller than 0.05, indicating significant correlations between the network features and application QoE.

C. QOE ML MODELING FOR 3 APPLICATIONS

1) DATA COLLECTION/PREPARATION

The data set is composed of descriptive features (link metrics) and a target feature (application QoE). The collection of link metrics relies on the Ryu controller's metrics collection function model/API, as detailed in Section II-C2. The QoE metrics for each application, including PESQ, APSNR, and Loading time, will be measured and calculated according to the methodology presented in Section II-B, serving as the raw data.

Preprocessing Stage: Before training, the raw data must be preprocessed to enhance its quality:

1. Handling Missing Values: Missing values may occur due to data corruption or collection failure, such as when Ryu fails to connect to switches. These values are removed.

2. Outlier Detection and Handling: Outliers can significantly impact model accuracy and must be handled appropriately. The outliers are detected and then removed before training.

3. Normalization: Given the large variance in feature scale in the raw dataset, normalization using Z-Score transform features to a similar scale, enhancing model performance and training stability and reducing training time.

2) DATASET FOR QOE MODELS TRAINING

After preprocessing, the data contains approximately 76,000 prepared samples, with each sample consisting of network metrics as descriptive features, and the corresponding application's QoE metrics, such as PESQ, APSNR, or Web map PLT, as the target feature. Here is the descprition of each dataset:

VoIP: The dataset consists of 8000 samples, with data features including link metrics (Delay, bandwidth, packet loss), and the target variable is PESQ.

Video streaming: The dataset comprises 32,000 samples, with data features including link metrics (Delay, bandwidth, packet loss). The target variable is APSNR.

Web map: The dataset comprises 36000 samples, with data features including link metrics (Delay, bandwidth, packet loss). The target variable is PLT.

3) TRAINING

A QoE prediction model is required for each application and each path length (2-link, 3-link, and 4-link). Four different ML algorithms — RF, MLR, SVR, and GBR — are employed to train the QoE prediction models for the three applications of VoIP, video, and Web maps. The evaluation results of the models after 5-fold cross-validation are presented in Table 3 in the subsequent section.

4) TRAINED MODELS EVALUATION

The Table 3 displays the performance of four ML algorithms (RF, MLR, SVR, GBR) across three applications (VoIP, Video, Web Map) using three metrics (MAE, RMSE, R^2) over 2-link, 3-link, and 4-link paths.

VoIP: RF and GBR excel with similar results, while MLR lags significantly, and SVR offers intermediate performance.

Video: RF slightly outperforms GBR, MLR provides poor results, and SVR demonstrates intermediate performance.

Web Map: Again, RF and GBR lead, while MLR underperforms, and SVR offers an intermediate result.

In summary, RF and GBR consistently present better performance than the other two algorithms, particularly in building with non-linear relationships, as revealed in the loading time of a web map. If packet loss is severe, the TCP connection may fail, resulting in an automatic reconnection and significantly increased loading time, deviating from a linear relationship. MLR performs the worst, possibly due to its inability to capture these non-linear relationships. Despite its suitability for small data, SVR is not appropriate for large data training (with over 10,000 samples in this case), resulting in suboptimal predictions. RF and GBR are somewhat affected by outliers and extended training time, but all models are trained under three minutes, with RF taking the longest.

Since real-time processing is not enforced, the offline training model allows for flexibility in choosing between RF and GBR, both of which provide the best prediction results. By comparing the average values of these Evaluation Metrics, RF presents a slightly better performance than the GBR. For routing experiments, therefore, RF is selected as the model used for QoE prediction when forecasting the QoE of a path.

D. QOE AND OTHER ROUTING STRATEGIES' PERFORMANCE ON SELECTING BEST PATH

A performance comparison was conducted between QoE routing and five other routing strategies: Bandwidth-based routing (BW_routing), delay-based routing (Delay_routing), packet loss-based routing (PL_routing), Bandwidth Delay Constrained routing (BW-Delay_routing), and Bandwidth Packetloss Constrained routing (BW-PL_routing). BW_routing, Delay_routing, and PL_routing select paths based on the highest bandwidth, shortest delay, or lowest packet loss. The algorithms of BW-Delay_routing and BW-PL_routing are presented in Section III-B.

Three hundred tests were conducted for each of the three applications, employing the same six routing algorithms and comparing all routing metrics under identical network link conditions. To evaluate the success of a routing strategy, the following process is presented:

1) Prior to implementing any routing strategy, route the application traffic through all three available paths and record the actual KPIs for each path. Identify the path with the highest QoE (KPI) on actual KPIs and label it as (*path_hQoE*);
2) Implement the routing strategies and calculate the predicted KPIs for each path. Compare the predicted KPIs of all three paths and select the one with the highest KPI as (*path_selected*), representing the path with the predicted highest QoE;
3) If *path_hQoE* matched *path_selected*, the routing strategy was deemed successful.

1) BEST-PATH SELECTION PERCENTAGE RESULT

The percentage of cases where the selected path matched the best path is calculated. Fig. 7 presents the Best-path Selection Percentages for the three applications using the six routing strategies. In Fig. 7, the y-axis indicates the best-path selection percentage. The higher the best-path selection percentage score, the more the selected path matches the best path, indicating that the chosen path has the highest QoE, referred to as the best-path. The figure reveals that QoE routing outperforms the other five strategies across three applications, showing the highest percentage of best paths selected, with a notable value of 0.7841.

The results demonstrate significant improvements achieved by the proposed QoE routing strategy in selecting the best path, outperforming all six evaluated algorithms. Specifically, QoE routing achieved the best path selection for VoIP in 80.07% of cases, video in 77.49%, and Web map in 77.66%. These figures compare favorably to the second-best routing strategy from existing techniques, which achieved 56.45%, 64.83%, and 58.6% respectively. On average, QoE routing selected the optimal path 78.41% of the time, in contrast to 58.6% by the next best strategy, BW-PL routing.

Furthermore, a close observation highlights that the other five routing strategies are primarily influenced by the most critical application-related network parameters, respectively. This is related to the results in Fig. 6, presenting the most critical network parameter related to each application. For example, VoIP is highly sensitive to packet loss, making it a critical network parameter. As illustrated in Fig. 7, packet loss emerges as the key factor contributing to a higher selection percentage for strategies like PL routing and BW-PL routing. Therefore, for video streaming, routing strategies related to bandwidth and packet loss (BW_routing, PL_routing, BW-PL_routing, and BW-Delay_routing) perform well. On the other hand, for web map routing, strategies related to delay (Delay_routing and BW-Delay_routing) perform best. This observation aligns with section IV-B, discussing the varied impact of network parameters on different applications.

2) BEST-PATH SELECTION CDF RESULT

To underscore a significant improvement in the QoE achieved by the QoE routing compared to alternative approaches, Fig. 8 portrays a Cumulative Distribution Function (CDF) comparison of optimal path selections across multiple routing strategies. CDF in Equation (4), where $F_X(x)$ represents the probability that X falls within the interval $[-\infty, x]$, and R represents the total range of x.

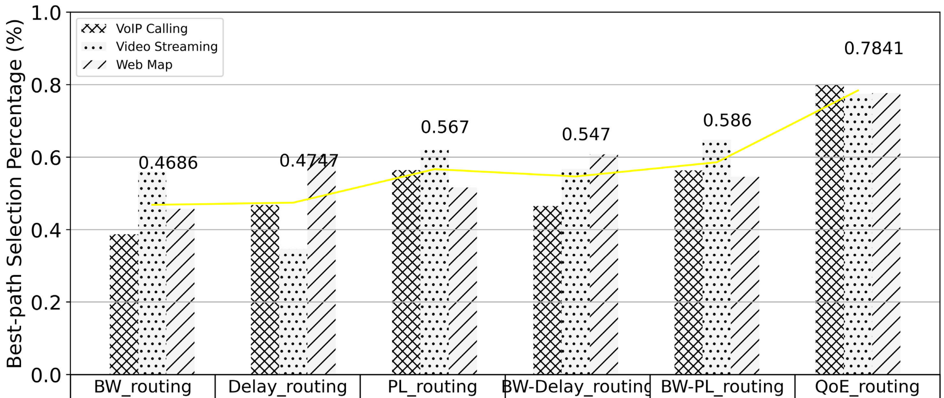
The equation to calculate CDF

$$F_X(x) = P(X \leq x), x \in R \tag{4}$$

The X-axis in Fig. 8 represents the QoE values, while the Y-axis indicates the probability that the QoE of the chosen path is less than or equal to a given value. In Fig. 8(a), for

TABLE 3. Multiple ML algorithms trained QoE models of 3 applications.

		2-link Path			3-link Path			4-link Path			Average		
		MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
VoIP	RF	0.3500	0.4808	0.7219	0.4538	0.5857	0.5377	0.5223	0.6819	0.5147	0.4421	0.5834	0.5914
	MLR	0.6819	0.9248	0.1320	0.6730	0.8813	0.0201	0.6602	0.8898	0.1632	0.6717,	0.8986,	0.1051
	SVR	0.5757	0.8205	0.3216	0.6140	0.8198	0.1720	0.6137	0.8452	0.2518	0.6123	0.8211	0.2456
	GBR	0.3635	0.4855	0.7206	0.4616	0.5957	0.5225	0.5213	0.6760	0.5251	0.4533	0.5923	0.5874
Video	RF	0.2468	0.4207	0.7936	0.2036	0.2963	0.8542	0.2293	0.3224	0.7833	0.2266	0.3465	0.8104
	MLR	0.6074	0.7585	0.2375	0.6463	1.006	-0.6458	0.7373	0.9300	-0.5727	0.6637	0.8982	-0.3270
	SVR	0.3086	0.4846	0.7226	0.2940	0.4274	0.7083	0.5978	0.8752	-0.2279	0.4001	0.5957	0.4010
	GBR	0.2744	0.4445	0.7720	0.2286	0.3151	0.8373	0.2815	0.3602	0.7431	0.2615	0.3733	0.7841
Web Map	RF	0.4854	0.6061	0.5008	0.4912	0.6402	0.6759	0.5268	0.4594	0.4124	0.5011	0.5686	0.5297
	MLR	0.5604	0.7399	0.2560	0.5524	0.7813	0.1821	0.6343	0.8492	0.0707	0.5824	0.7901	0.1696
	SVR	0.5272	0.6846	0.3631	0.5553	0.7879	0.1662	0.5911	0.7973	0.1775	0.5579	0.7566	0.2356
	GBR	0.4803	0.5964	0.5168	0.4826	0.6240	0.4817	0.5065	0.6556	0.4503	0.4898	0.6252	0.4829



	BW_routing	Delay_routing	PL_routing	BW-Delay_routing	BW-PL_routing	QoE_routing
VoIP Calling	0.3874	0.4686	0.5645	0.4652	0.5641	0.8007
Video Streaming	0.5608	0.3468	0.6199	0.5678	0.6483	0.7749
Web Map	0.4575	0.6088	0.5166	0.6081	0.5457	0.7766
Average	0.4686	0.4747	0.567	0.547	0.586	0.7841

FIGURE 7. Best-path selection percentage of multiple routing strategies.

example, the black-circled point on the purple line indicates that when the QoE routing strategy is applied to select the best path, there is a probability of approximately 0.6 that the PESQ score achieved by this chosen path is less than 3.0. In this context, CDF is employed to assess and compare the QoE achieved by each selected path using various routing strategies.

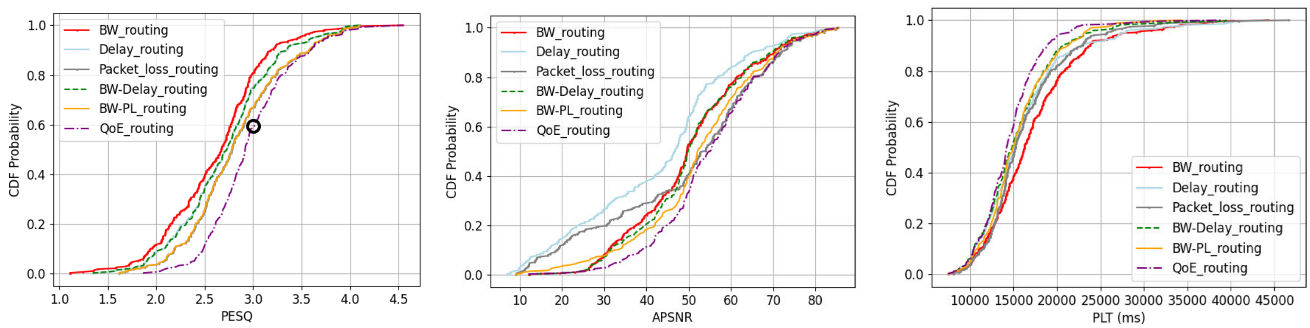
In Fig.8(a), six CDF lines represent different routing strategies. The more the line is shifted to the right, it indicates that the PESQ achieved by the corresponding routing is distributed in a higher PESQ value range. Naturally, for VoIP, higher PESQ values correspond to better performance. The purple CDF line for QoE routing is shifted to the right, indicating that it consistently achieves the highest PESQ score for the selected path. This means that, given an equal probability, the QoE routing (purple line) is more likely to result in the highest PESQ path. For instance, at a probability of 0.5, the QoE routing selects a path with a PESQ score of < 2.9, while other routing strategies yield selects paths with PESQ scores of 2.8, 2.8, 2.75, 2.75 and 2.6 respectively. The order of shifting, from best to worst, is: QoE, BW-PL/Packetloss, BW-Delay/Delay, BW routing.

Therefore, in the case of VoIP, at equivalent probability levels, QoE routing consistently prioritizes the path with the highest PESQ score when compared to other routing strategies. This comparison extends to the other two applications, where the CDF results indicate that, under the same probability conditions, the path selected by QoE routing attains the highest KPIs score.

E. QOE AND OTHER ROUTING STRATEGIES PERFORMANCE ON QOE DIFFERENCE

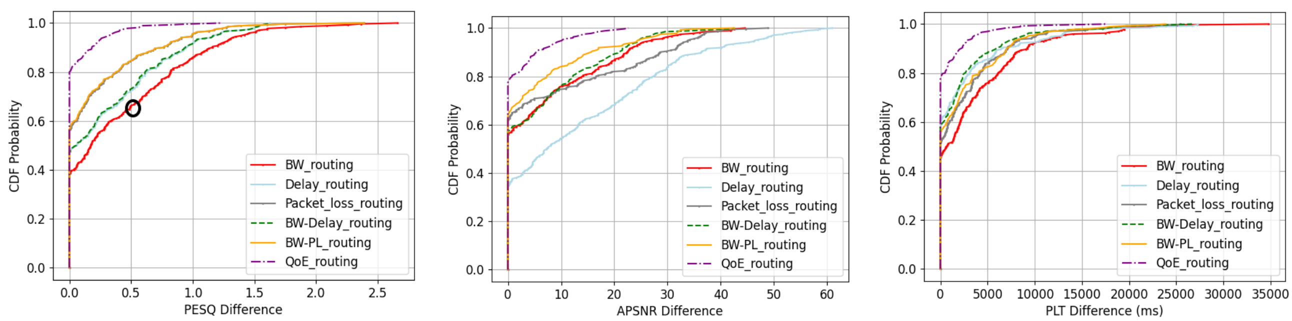
Selecting the best path based on percentage alone, however, does not provide a comprehensive understanding of the effect on routing decisions in the network. Often, two competing paths may exhibit very similar KPIs and QoE scores, making the choice between them less consequential. For instance, if the video QoE APSNR on path 1 is 76, and path 2 is 75, with path 3 at 40, path 1 will be chosen for its highest QoE. However, the small APSNR difference between paths 1 and 2 implies that either would offer an almost identical user experience.

In such scenarios, it's more informative to examine the QoE differences between the chosen path and the path with



(a) CDF of Best-path Selected for VoIP (b) CDF of Best-path Selected for Video (c) CDF of Best-path Selected for Web Map

FIGURE 8. CDF analysis comparing best-path selected across various routing strategies.



(a) VoIP CDF (b) Video CDF (c) Web Map CDF

FIGURE 9. Comparison of QoE difference for multiple routing strategies.

the highest QoE, as defined in Equation (5). A smaller QoE difference signifies that the selected routing's QoE is closer to the best possible, thus indicating a more optimal strategy. This relationship can be illustrated with a Cumulative Distribution Function (CDF) in Equation (4), where $F_X(x)$ represents the probability that X falls within the interval $[-\infty, x]$, and R represents the total range of x. By applying the CDF to the QoE differences (QoE_{diff}), $F_{QoE_{diff}}(x)$ represents the probability that QoE_{diff} is smaller than the value x.

The equation to calculate QoE Difference

$$QoE_{diff} = QoE_{best} - QoE_{selectedPath} \tag{5}$$

In Fig. 9, the X-axis represents the value of the QoE_{diff}, and the Y-axis represents the probability that the QoE_{diff} is less than or equal to that value. Taking Fig. 9(a) as an example, where $PESQ_{diff}$ denotes the difference between the highest and achieved PESQ, the point circled in black represents that the probability of $PESQ_{diff}$ being less than 0.5 is approximately 0.65. In this context, CDF is used to compare QoE_{diff} across routing strategies.

- VoIP: As illustrated in Fig. 9(a), six CDF lines represent the $PESQ_{diff}$ for different routing strategies. QoE routing (purple CDF) is left-shifted the furthest, indicating the smallest $PESQ_{diff}$. The order of shifting, from best to worst, is: QoE, BW-PL/Packetloss, BW-Delay/Delay, BW. The 75th percentiles corroborate this order, with values of 0, 0.3, 0.6, and 0.75 respectively. QoE routing consistently offers optimal performance for VoIP;
- Video: In Fig. 9(b), QoE routing provides the smallest $APSNR_{diff}$, followed by BW-PL, BW-Delay, Packetloss, BW, and Delay routing. The 75th percentiles further highlight QoE routing's superior performance, with values from 0 to 25. Again, QoE routing excels in video, as observed in VoIP;
- Web map: In Fig. 9(c), the QoE routing's superiority continues, with a shift rate and slope scale order: QoE, BW-Delay, Delay, BW-PL, and BW-routing. The 75th percentile PLT_{diff} further supports this pattern, with values of 0, 2500, 2510, 2500, 3000, and 5000 respectively.

In Fig.9, it is evident that the QoE difference exhibits the smallest variation for QoE routing compared to the other routing strategies. In the case of VoIP, the order of QoE difference CDF lines precisely mirrors the reverse order of the Best-path selection CDF (Fig.8). This indicates a consistent performance ranking for all six routing strategies across both evaluation aspects. Similar results are observed for the other two applications, highlighting a consistent evaluation outcome. Consequently, the findings from both best-path selection and QoE difference aspects align, ultimately demonstrating that QoE routing outperforms the other five routing strategies.

The results presented in IV-D and Section IV-E illuminate the efficacy of different routing strategies concerning the number of accurately selected paths and the QoE difference away from the best path. Notably, QoE routing consistently emerges as the optimal strategy, surpassing the others across all tested applications.

V. CONCLUSION

This paper has explored the efficacy of a QoE-oriented routing strategy for various applications, leveraging network link metrics. The framework, utilizing SDN tools, constructs a relationship model between the network link and application QoE through ML algorithms. Through rigorous testing against five alternative strategies, the proposed QoE routing strategy consistently outperforms, achieving the best path selection in 80.07% of VoIP cases, 77.49% for video, and 77.66% for Web map applications, an average improvement of 22% over the second-best method. An examination of the QoE difference CDF further demonstrates that the QoE routing chooses paths closer to the optimal QoE, providing a nearly optimal performance. This research not only emphasizes the significance of QoE routing but also presents a strategy that can notably elevate application performance, opening doors for a responsive QoE feedback model.

The QoE routing methodology, initially validated in a simulated Mininet network, holds promise for real-world implementation in live networks featuring actual devices and traffic, thereby enhancing its practical significance. Additionally, the reliance on a ML-based QoE model suggests potential for improvement through the integration of authentic human feedback a consideration for future development. Our approach involves incorporating RL in the training process to refine the model's adaptability and precision. Real-time QoE feedback from end-users is collected and compared to predicted KPIs. Higher rewards are assigned when user feedback closely aligns with predicted KPIs, facilitating model fine-tuning through RL algorithms and adaptive decision-making based on user input. This dynamic process contributes to improved overall performance and user satisfaction. Despite the ongoing need for refinement, our QoE routing strategy demonstrates noteworthy strengths by prioritizing user experience over conventional network service enhancements. Its practical applicability extends beyond the current context, making it suitable for diverse scenarios and establishing our research as a fundamental contribution to the evolution of QoE routing. Looking ahead, the findings from this study provide a foundation for future endeavors aimed at developing an optimized and adaptive QoE routing system.

REFERENCES

[1] G. R. Ash, R. H. Cardwell, and R. P. Murray, "Design and optimization of networks with dynamic routing," *Bell Syst. Tech. J.*, vol. 60, no. 8, pp. 1787–1820, Oct. 1981.

[2] X. Liu, Y. Wang, and Y. Liu, "QFR: A QoE-driven fine-grained routing scheme for virtual reality video streaming over SDN," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, May 2020, pp. 1–6.

[3] O. Lemeshko, M. Yevdokymenko, O. Yeremenko, A. M. K. Al-Dulaimi, P. Segec, and J. Papán, "QoE-centric adaptive multipath routing under R-factor with load balancing in SDN," in *Proc. IEEE Int. Conf. Problems Infocommun., Sci. Technol.*, Oct. 2020, pp. 91–96.

[4] L. Wang, A. Durning, and D. T. Delaney, "ML-based video streaming QoE modeling with E2E and link metrics," in *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2022, pp. 1–8.

[5] L. Wang and D. T. Delaney, "QoE oriented cognitive network based on machine learning and SDN," in *Proc. IEEE 11th Int. Conf. Commun. Softw. Netw. (ICCSN)*, Jun. 2019, pp. 678–681.

[6] E. S. Crawley, R. Nair, D. B. Rajagopalan, and H. J. Sandick, *A Framework for QoS-based Routing in the Internet*, document RFC 2386, 1998. [Online]. Available: https://www.rfc-editor.org/info/rfc2386

[7] R. Bellman, "On a routing problem," *Quart. Appl. Math.*, vol. 16, no. 1, pp. 87–90, 1958.

[8] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.

[9] W. Zhengying, S. Bingxin, and Z. Erdun, "Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm," *Comput. Commun.*, vol. 24, nos. 7–8, pp. 685–692, Apr. 2001.

[10] S. Tomovic and I. Radusinovic, "Fast and efficient bandwidth-delay constrained routing algorithm for SDN networks," in *Proc. IEEE NetSoft Conf. Workshops (NetSoft)*, Jun. 2016, pp. 303–311.

[11] J. W. Guck, M. Reisslein, and W. Kellerer, "Function split between delay-constrained routing and resource allocation for centrally managed QoS in industrial networks," *IEEE Trans. Ind. Informat.*, vol. 12, no. 6, pp. 2050–2061, Dec. 2016.

[12] Q. Ma and P. Steenkiste, "On path selection for traffic with bandwidth guarantees," in *Proc. Int. Conf. Netw. Protocols*, 1997, pp. 191–202.

[13] A. F. Khalifeh and A. H. El-Mousa, "QoS routing of VoIP using a modified widest-shortest routing algorithm," in *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl.*, May 2007, pp. 116–123.

[14] D. Medhi and K. Ramasamy, *Network Routing: Algorithms, Protocols, and Architectures*, 2nd ed., San Francisco, CA, USA: Morgan Kaufmann, 2017.

[15] V. Tejaswini, M. Sayeekumar, and G. M. Karthik, "Fast bandwidth-delay routing methods for software-defined network (SDN)," in *Artificial Intelligence Techniques for Advanced Computing Applications*, D. J. Hemanth, G. Vadivu, M. Sangeetha, and V. E. Balas, Eds., Singapore: Springer, 2021, pp. 323–334.

[16] J. Gong and A. Rezaeipanah, "A fuzzy delay-bandwidth guaranteed routing algorithm for video conferencing services over SDN networks," *Multimedia Tools Appl.*, vol. 82, no. 17, pp. 25585–25614, Jul. 2023.

[17] H.-W. Kao and E. H. Wu, "QoE sustainability on 5G and beyond 5G networks," *IEEE Wireless Commun.*, vol. 30, no. 1, pp. 118–125, Feb. 2023.

[18] J. Nightingale, P. Salva-Garcia, J. M. A. Calero, and Q. Wang, "5G-QoE: QoE modelling for ultra-HD video streaming in 5G networks," *IEEE Trans. Broadcast.*, vol. 64, no. 2, pp. 621–634, Jun. 2018.

[19] H. Nam, K.-H. Kim, J. Y. Kim, and H. Schulzrinne, "Towards QoE-aware video streaming using SDN," in *Proc. IEEE Global Commun. Conf.*, Dec. 2014, pp. 1317–1322.

[20] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "A novel QoE-centric SDN-based multipath routing approach for multimedia services over 5G networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.

[21] J. Yan, J. Li, Y. Fang, Z. Che, X. Xia, and Y. Liu, "Subjective and objective quality of experience of free viewpoint videos," *IEEE Trans. Image Process.*, vol. 31, pp. 3896–3907, 2022.

[22] Z. Hu, H. Yan, T. Yan, H. Geng, and G. Liu, "Evaluating QoE in VoIP networks with QoS mapping and machine learning algorithms," *Neurocomputing*, vol. 386, pp. 63–83, Apr. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231219317813

[23] T. Jie, X. Yan, C. Yali, and L. Xuhui, "Multi-path QoE routing algorithm for wireless mesh network based on Q-learning," *J. Phys., Conf. Ser.*, vol. 2384, no. 1, Dec. 2022, Art. no. 012040.

[24] Z. Wang and J. Crowcroft, "Bandwidth-delay based routing algorithms," in *Proc. GLOBECOM*, 1995, pp. 2129–2133.

[25] Q. Ma and P. Steenkiste, "Quality-of-service routing for traffic with performance guarantees," in *Building QoS Into Distributed Systems*. Cham, Switzerland: Springer, 1997, pp. 115–126.

[26] L. Lazic and N. Mastorakis, "Cost effective software test metrics," *WSEAS Trans. Comput.*, vol. 7, no. 6, pp. 599–619, 2008.

[27] *Series G: Transmission Systems and Media, Digital Systems and Networks*, Standard, Telecommunication Standardization Sector of ITU, 2000.

[28] A. Munaretto, H. Badis, K. Al Agha, and G. Pujolle, "A link-state QoS routing protocol for ad hoc networks," in *Proc. 4th Int. Workshop Mobile Wireless Commun. Netw.*, 2002, pp. 222–226.

[29] C. Thorpe, A. Hava, J. Langlois, A. Dumas, and C. Olariu, "IMOS: Enabling VoIP QoS monitoring at intermediate nodes in an OpenFlow SDN," in *Proc. IEEE Int. Conf. Cloud Eng. Workshop (IC2EW)*, Apr. 2016, pp. 76–81.

[30] L. Wang and D. T. Delaney, "ML-based application performance modelling over a SDN network using end-to-end and link metrics," in *Proc. Eur. Wireless; 26th Eur. Wireless Conf.*, Nov. 2021, pp. 1–8.

[31] G. Kakkavas, A. Stamou, V. Karyotis, and S. Papavassiliou, "Network tomography for efficient monitoring in SDN-enabled 5G networks and beyond: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 59, no. 3, pp. 70–76, Mar. 2021.

[32] X. Zuo, J. Yang, M. Wang, and Y. Cui, "Adaptive bitrate with user-level QoE preference for video streaming," in *Proc. IEEE Conf. Comput. Commun.*, May 2022, pp. 1279–1288.

[33] G. Kougioumtzidis, V. Poulkov, Z. D. Zaharis, and P. I. Lazaridis, "A survey on multimedia services QoE assessment and machine learning-based prediction," *IEEE Access*, vol. 10, pp. 19507–19538, 2022.

[34] T. Abar, A. Ben Letaifa, and S. E. Asmi, "Machine learning based QoE prediction in SDN networks," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2017, pp. 1395–1400.

[35] *Qoe and Networking Community*. Accessed: Feb. 1, 2024. [Online]. Available: http://qoe.community/

[36] *Mininet, an Instant Virtual Network on Your Laptop (or Other PC)*. Accessed: Feb. 1, 2024. [Online]. Available: http://mininet.org/

[37] A. Durning and L. Wang. (2022). *Qoe Routing Module for Ryu*. Accessed: Feb. 1, 2024. [Online]. Available: https://github.com/adam-durning/ryu/tree/development_branch/ryu/app/qoe_app

[38] J. Y. Yen, "Finding the K shortest loopless paths in a network," *Manage. Sci.*, vol. 17, no. 11, pp. 712–716, Jul. 1971.

[39] R. Schatz, T. Hoßfeld, L. Janowski, and S. Egger, "From packets to people: Quality of experience as a new measurement challenge," in *Data Traffic Monitoring and Analysis*. Cham, Switzerland: Springer, 2013, pp. 219–263.

[40] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2001, pp. 749–752.

[41] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electron. Lett.*, vol. 44, no. 13, p. 800, 2008.

[42] R. Rassool, "VMAF reproducibility: Validating a perceptual practical video quality metric," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2017, pp. 1–2.

[43] A. Horé and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 2366–2369.

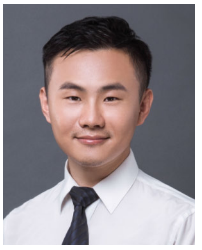
[44] A. Chan, K. Zeng, P. Mohapatra, S.-J. Lee, and S. Banerjee, "Metrics for evaluating video streaming quality in lossy IEEE 802.11 wireless networks," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[45] H. Z. Jahromi, D. T. Delaney, and A. Hines, "Beyond first impressions: Estimating quality of experience for interactive web applications," *IEEE Access*, vol. 8, pp. 47741–47755, 2020.

[46] H. Z. Jahromi, D. T. Delaney, B. Rooney, and A. Hines, "Establishing waiting time thresholds in interactive web mapping applications for network QoE management," in *Proc. 30th Irish Signals Syst. Conf. (ISSC)*, Jun. 2019, pp. 1–7.

[47] H. Z. Jahromi, A. Hines, and D. T. Delanev, "Towards application-aware networking: ML-based end-to-end application KPI/QoE metrics characterization in SDN," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2018, pp. 126–131.

LEI WANG received the M.Sc. degree in telecommunication engineering from the University of Birmingham, Birmingham, U.K., in 2016. She is currently pursuing the Ph.D. degree with the Department of Electrical and Electronic Engineering, University College Dublin, Dublin, Ireland. Her research interests include network data analytics, optimization of user QoE, and adaptable programmable networks.



XIPING WU (Senior Member, IEEE) received the Ph.D. degree from The University of Edinburgh, U.K., in 2015. He is currently an Assistant Professor with the School of Electrical and Electronic Engineering, University College Dublin (UCD), Ireland. Prior to joining UCD, he was a Postdoctoral Research Associate with the University of Oxford. His research interests include 6G mobile communications, optical wireless communications (OWC), and the Internet of Things, with a particular focus on developing hybrid wireless networks that integrate OWC and radio frequency, empowered by software-defined networking, and artificial intelligence. He has authored or coauthored over 50 publications in these areas. He was a recipient of the Best Paper Award at IEEE ICCC, in 2021, and the Royal Irish Academy (RIA) Charlemont Grant Award, in 2022.

DECLAN T. DELANEY received the Ph.D. degree in network analysis and design for the IoT from the School of Computer Science, University College Dublin (UCD), Dublin, Ireland, in 2015. He was with LMI Ericsson, Dublin, and has collaborated with SMEs on Horizon Europe funding proposals. He is currently an Assistant Professor with the School of Electrical and Electronic Engineering, UCD. He is an SFI Funded Investigator of the Project CONSUS (https://www.ucd.ie/consus), an SFI industry-funded collaboration focused on precision agriculture; a Principal Investigator of the EPA-funded SmartBOG Project (https://www.smartbog.com); and a Principal Investigator of the CAMEO Project (https://www.cameoplatform.ie). His research interests include network data analytics for adaptable programmable networks and infrastructure and data assurance for the IoT and sensor systems.