sciendo

Transport and Telecommunication, 2025, volume 26, no. 3, 237-249
Transport and Telecommunication Institute, Lauvas 2, Riga, LV-1019, Latvia
DOI 10.2478/ttj-2025-0018

QUEUING DELAY REDUCTION BASED ON NETWORK TRAFFIC PATTERNS: A PREDICTIVE QOS FRAMEWORK FOR POINT-TO-POINT COMMUNICATIONS

Albert Espinal[1], V. Sanchez Padilla[2, 3]

[1] Escuela Superior Politécnica del Litoral, Telematics Engineering Dept., Guayaquil, Ecuador
aespinal@espol.edu.ec
[2] Virginia Polytechnic Institute and State University, Engineering Education Dept., Blacksburg, VA, USA
[3]Universidad ECOTEC, College of Engineering, Architecture and Natural Sciences, Samborondón, Ecuador
vsanchez@vt.edu, vlsanchez@ecotec.edu.ec

Ensuring optimal quality of service (QoS) in computer networks requires a detailed assessment of performance metrics, with data network queuing delay within intermediate devices being critical parameters. This paper presents a predictive Quality of Service (QoS) model designed to reduce queuing delays by analyzing traffic patterns in intermediate devices in point-to-point network connections. The proposed novel Length Packet Queuing (LPQ) model leverages packet length analysis to predict and manage queuing delays without relying on traditional packet marking mechanisms. Through Poisson distribution and polynomial regression models, network traffic patterns and queuing delays are estimated, respectively, demonstrating significant improvements of conventional QoS models. Simulations and experimental scenarios validated the LPQ model's effectiveness, showing lower delays through various network loads and traffic conditions. The results of this research highlight the potential of the novel LPQ model for enhancing QoS in hybrid networks, where user applications generate diverse packets.

Keywords: Quality of service, queuing delay, network delay, network traffic patterns, point-to-point links

1. Introduction

Internet users expect to access network content instantly for various purposes, including video conferences, streaming, and academic queries. Data traffic can sometimes exceed the link bandwidth between the transmitter (e.g., provider) and the receiver (e.g., end-user) when conveying information. Therefore, quality of service (QoS) in data networks must guarantee data transmissions through network designs that involve QoS tools. These tools ensure that specific types of content, such as voice or video, have transmission priority over best-effort traffic, usually related to e-mail and web browsing navigation (Narayanaswamy & Rajan, 2021). Devices typically implement QoS configurations during traffic congestion, where packet classification in software queues and packet scheduling for dispatch affect queuing delay (Sumarsono & Rodriguez, 2021).

QoS concepts apply to a variety of network technologies and services. The literature presents extensive studies aimed at improving QoS in delay-sensitive applications, such as Voice and Video over IP or streaming (Fiedler *et al*., 2010; Kempa, 2013; Yihunie & Abdelfattah, 2018) to enhance network performance. Other research focuses on guaranteeing QoS in computer networks communicating through mobile ad hoc networks (Wenbin *et al.*, 2017), LTE (Gómez *et al.*, 2014), NGN (Ghazel & Saïdane, 2015), CDMA (Vassilakis *et al.,* 2018), and wireless mesh networks (Gheisari *et al*., 2020) ensuring reliability and mitigating delay issues. Nevertheless, these works do not associate their models with packet length patterns.

Networking traffic requirements for voice, video, and data must be managed differently. Voice traffic requires predictable bandwidth and consistent packet arrival times, making it sensitive to delays and packet losses (Daza Alava *et al*., 2021). Traffic parameters should not exceed defined thresholds to maintain acceptable service quality. For optimal data network service, voice and jitter should not exceed 150 milliseconds and 30 milliseconds, respectively, with packet loss at 1% or below (Daza Alava *et al*., 2021; Di Mauro & Liotta, 2020). Video transmission traffic tends to be unpredictable, inconsistent, and bursty (Lindeberg *et al.*, 2011). Video is typically less resilient to losses and involves large data volumes per packet, with data traffic striving for real-time transmission and unpredictable bandwidth demands (Alaya

et al., 2021). Most data applications use TCP (Transmission Control Protocol), to a lesser extent using UDP (User Datagram Protocol). The networking traffic generated by applications such as YouTube and social media comprises most of the Internet traffic, predominantly consisting of streaming and interactive content that relies on TCP (Hodroj *et al*., 2021).

Traffic classification involves automated methods to analyze and classify data network packet sets based on traffic features according to specific criteria. Classifying traffic is essential in computer networks for design, resource provisioning, security assurance, trend analysis, and shaping QoS techniques. Studies carried out by Espinal *et al*. (2019a, 2019b, 2019c, 2020) analyzed traffic generated by electronic devices, including desktop computers, laptops, and mobile phones in scenarios involving a hybrid campus network and an LTE cellular network. In these works, the analyzes that involve wireless and Ethernet communication used a novel sniffer that discarded packet payloads and only retrieved the packet headers, while the packets generated through mobile communication were based on a proprietary tool. The authors emphasized the contribution of the most common protocols and applications. Statistical models, including Poisson distributions, allow for network traffic evaluation through the representation of stochastic behavior. These models are suitable for analyzing one-way queuing delays and complement the predictive QoS model proposed in the present research.

Upon transmitting data from a source host to a receiver device, network packets experience a certain degree of delay. Network policies, such as QoS and packet filtering, address delay issues (Floyd, 2008). End-to-end network delay measures the time from when data is created by an application and delivered to an operating system, passed through a network interface card (NIC) for encoding, transmitted through a physical medium, received, and forwarded by intermediate devices until reaching the destination. This leads data networks to undergo queuing delays, which must be considered in the sizing of an overall network delay. Queuing delay refers to the time a packet remains in transmission queues after being sent by the source host (Chefrour, 2021). Queues help in preventing packet losses; however large queue sizes can cause considerable delays. Controlling packet congestion is one way to tolerate queuing delays. Packet delay in a queue depends on the number of packets that arrive beforehand and those already waiting for transmission over the medium. Packet delays vary significantly from one packet to another depending on the traffic type and intensity.

Previous studies have analyzed Round Trip Time (RTT) and One-Way Delay (OWD) using various topologies and methods to simulate network traffic and estimate models of queuing delay, the most significant component for delay calculations. Some studies propose delay estimation methods using clock synchronization mechanisms to determine the precision and sensitivity of queuing delay (Ferencz & Kovacshazy, 2014; Liu, 2014; Kompella *et al.*, 2012; Salehin *et al.,* 2019). Other tests examine the variable and constant components of one-way delay (Csoma *et al*., 2015; Sukhov *et al*., 2016; Ulbricht & Wagner, 2016). Some of these studies employ synchronization mechanisms for accurate sampling but at a higher cost related to hardware deployments. However, traffic generated based on distributions is presented in Espinal *et al.* (2024), providing better queuing delay estimations based on polynomial regressions. In the experimental scenario, a more standardized synchronization protocol was used, such as the Network Time Protocol (NTP).

The primary goal of this work is to present the Length Packet Queuing (LPQ) model as a QoS predictive model that reduces queuing delay times compared to benchmark models. The proposed model is a variant of the DiffServ architecture but without marking and classifying the Differentiated Service Code Point (DSCP) field of IPv4 packets or the Traffic Class (TC) field of IPv6 packets. It analyzed OWD behavior using benchmark models, emphasizing queuing delay in a TCP/IP network with a point-to-point topology, and estimated trends using predictive models based on polynomial regression. This research is expected to contribute to the ongoing pursuit of optimizing network performance through predictive modeling with tangible benefits for point-to-point data connections.

2. Background

This work presents a predictive QoS model to minimize queuing delays by analyzing traffic patterns in intermediate devices in point-to-point connections based on three-stage research. In the initial phase, we identified prevalent packet sizes for contemporary protocols and applications in both wired and wireless environments, estimating this traffic as a Poisson distribution. Subsequently, the second stage delves into a comprehensive analysis of the components contributing to the end-to-end network delay with a particular emphasis on queuing delay. This phase employs polynomial regressions to model the behavior of queuing delay effectively. The third and final stage proposes the LPQ model as a predictive QoS model grounded in real-traffic parameters, proficient in estimating queuing delay. Numerical results attest to the efficacy of

this model, showing significantly lower queuing delay measures when compared to benchmark common models.

2.1. Traffic modelling (First phase)

The first stage involved a detailed study of the packet length variable for devices connected to wired, wireless, and mobile networks, the latter with a focus on an LTE (Long Term Evolution) network, with tests covering different protocols (IPv4, IPv6, TCP, UDP). These protocols were used in the context of common applications related to social media platforms (such as Facebook, Instagram) and email services, among others. The purpose of choosing packet length for proposing the current QoS model regards the need to explore other options to study queueing delay, as this parameter has not been examined in ways of its variability. Previous work carried out by one of the authors explained the convenience of going through this type of analysis. The aim was to characterize and model these variables to gather the necessary data by implementing various network scenarios based on the proposed convergent network topology. Millions of data packets were collected in each scenario and later analyzed to estimate their stochastic behavior using Poisson distributions. The results are presented in Espinal *et al*. (2019a, 2019b, 2019c, 2020).

The bimodal behavior observed in the tests conducted on a wired network using IPv4, TCP, and UDP is represented in Figure 1. The adjusted model consists of two Poisson distributions, with $\lambda_1 = 90.61$ and $\lambda_2 = 1458.72$. The probability that a packet length belongs to the first distribution is 0.469, while for the second distribution, it is 0.531. The sum of the two Poisson distributions forms the model shown in Equation 1. The development of the first stage made it possible to obtain the average length parameters of large, medium, and small packets (LPPG, LPPM, LPPP), along with their probabilities of occurrence (PrPG, PrPM, PrPP). These are considered input parameters for the LPQ model.

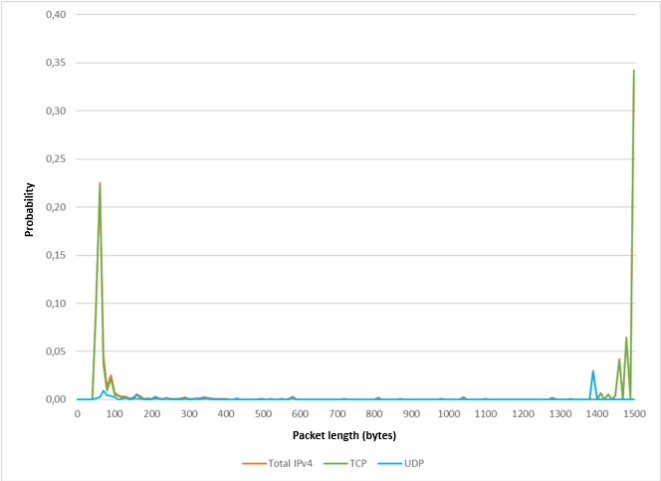


Figure 1. IPv4 wired network distribution traffic

$$P(X = x) = 0.469 * \frac{e^{-90.61}90.61^{x}}{x!} + 0.531 * \frac{e^{-1458.72}1458.72^{x}}{x!} \quad (1)$$

2.2. Estimation of the queuing delay (Second phase)

The second stage used the models and estimates from the first phase to study one-way and queuing delays. This study on OWD provides data on transmission, propagation, processing, and queuing delays. The focus was on the last type of delay, as it plays a key role in the management of data packets in output queues, regardless of whether they are linked to a quality-of-service policy. Based on the results of this phase, it became possible to model the predictive behavior of queuing delay using the polynomial regression method, as shown in Espinal *et al.* (2024). This method is compared with other regression techniques to determine which one best represents the relationship between the dependent and independent variables. It allows us to define a baseline for the expected times for queuing delay, contributing to its reduction through the LPQ model proposed in this study.

This proposal relies on the employment of Class-Based Weighted Fair Queuing (CBWFQ) to deal with packet features during congestion or simulation scenarios, especially when minimum bandwidth can be offered upon packet transmissions (Zakariyya & Rahman, 2015). Figure 2 illustrates the queuing delay

for the CBWFQ technique, with Equation 2 showing its estimation using a predictive model based on polynomial regression. The function $f(x)$ shown in (2) represents the average queuing delay, where x refers to the packet lengths. This phase provides the average queuing delay parameters REMPG, REMPM, and REMPP, which are listed along with the average length parameters and the respective probability of occurrence as inputs for the LPQ model. These models serve as a baseline in the third phase of the research for a comparative analysis with delays obtained from the simulation of the QoS LPQ model.

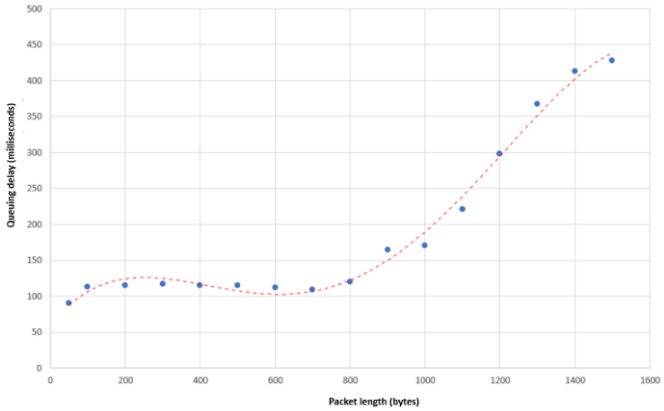


Figure 2. CBWFQ queuing delay analysis

$$f(x) = -7x10^{-10}\,X^4 + 2x10^{-6}\,X^3 - 0.002\,X^2 + 0{,}6351\,X + 60{,}683. \quad (2)$$

It was considered a networking link load to study the network delay. The link load simulated the experimental topology by the traffic estimation models analyzed in Espinal *et al.* (2019a, 2019b, 2019c, 2020) utilizing Poisson stochastic process modeling. For delay measurement, the generated traffic comprises packets with timestamp headers. These packets are synchronized using NTP processed at the measurement points, where the values of different delay components were collected. The queuing delay is modeled and compared with data from the QoS predictive model. To simulate an enterprise network, four logical segments (subnets) were configured to generate point-to-point serial link loads representing a connection to an Internet Service Provider (ISP). The traffic generated between the networks 192.168.1.0/24 and 192.168.3.0/24 allowed a load simulation on the point-to-point link based on actual protocol and application patterns, employing open-source and freely available tools. Figure 3 presents the network scenario used to measure the corresponding delays.

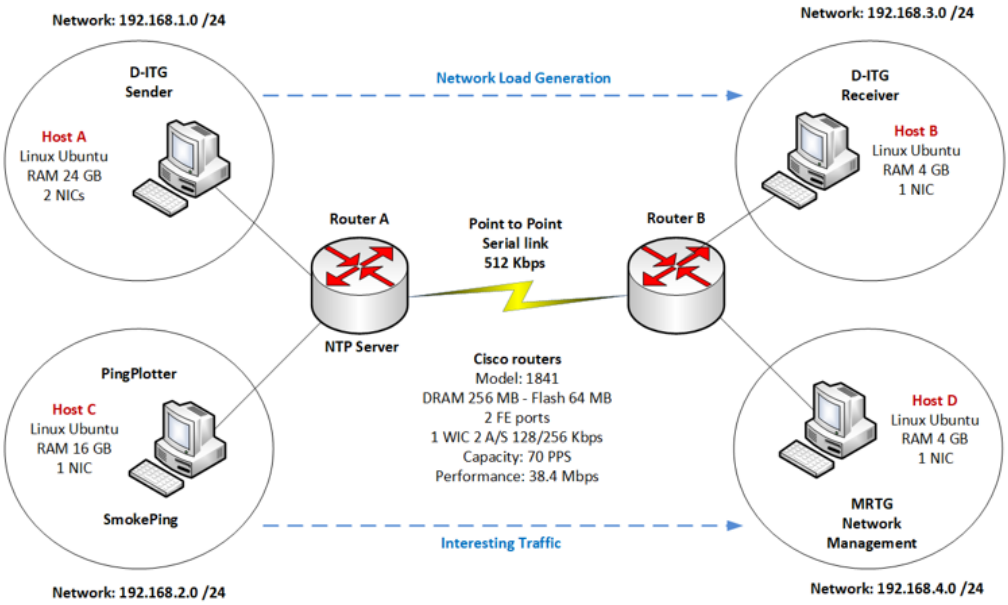


Figure 3. Experimental setting for delay measurement

A Distributed Internet Traffic Generator (D-ITG) was set up to generate network traffic (Botta *et al*., 2012) following the stochastic behavior of two random variables, the inter-departure time between packets (IDT), and the packet length. For traffic generation, the Poisson distribution parameters were configured in the D-ITG to convey variable-length packets with higher probabilities to the binomial points. It is significant to synchronize the equipment involved in the topology through NTP in server mode on one of the routers and in client mode on the other devices. Wireshark was employed only for packet capturing, specifically the headers for posterior analysis. A Multi Router Traffic Grapher (MRTG) based on Simple Network Management Protocol (SNMP) monitors and measures traffic and link loads. A CBWFQ QoS policy was configured in the routers to determine queuing delay values under different link load levels, such as 32, 64, 128, and 256 Kbps. A router was set up as an NTP server to synchronize all devices involved in topology.

Wireshark captured the packet headers and timestamps from the start point of the traffic interest towards the arrival point. Consequently, it was achievable to get the OWD with components estimated according to pre-established formulas. The Wireshark sniffer captured packets at the input and output of the routers to measure the OWD between the two devices as well. SmokePing and Ping Plotter were employed to validate measurements that covered OWD, RTT, jitter, and packet loss.

Weighted Fair Queuing (WFQ) ensures a fair distribution of packet flows passing through a link. This protocol uses layer 3 and 4 parameters of an IP packet to generate low, normal, medium, and high-priority queues. Its configuration can be fundamental using the fair-queue command on a defined interface. In the case of CBWFQ, the behavior is similar but through queues outlined by the user. For this case, the configuration can include an access list, class map, and policy map, all supported by CISCO IOS routers. A bandwidth percentage is assigned to them based on the number of packets expected according to their size, as defined in the First Phase of the proposal. In addition, the defined policy relates to an interface using a service policy in the direction in which the packets exit through it.

The CBWFQ QoS policy followed the four load levels ranging from 32 to 256 Kbps mentioned above. For each load level, it was configured 16 packet lengths, ranging from 50 to 1500 bytes for all traffic types. For each packet length, there were 11 different samples collected, and the median values were used for regression analysis. The propagation delay and processing delay during the tests for all packet lengths resulted in 0.0001 ms and 0.048 ms, respectively. Median values for WFQ with 256 Kbps load were used. Table 1 displays the OWD component values. Figures 4 and 5 present the results obtained for both OWD and queuing delay using CBWFQ QoS policy.

The data obtained were useful to estimate a predictive model for OWD and queuing delay based on polynomial regression (Equation 3). Table 2 presents the coefficients for the OWD, and queuing delay analysis model classified by load level. These models served as a reference for comparative analysis based on the delays predicted by the LPQ model.

Table 1. OWD delays through the experiment

Packet length (bytes)	Transmission delay (ms)	Queuing delay (ms)	OWD (ms)
50	3.13	13.819	17.00
100	6.27	14.986	21.30
200	12.53	19.120	31.70
300	18.80	23.354	42.20
400	25.06	28.238	53.35
500	31.33	21.972	53.35
600	37.60	35.356	73.00
700	43.86	39.940	83.85
800	50.13	41.774	91.95
900	56.39	60.208	116.65
1000	62.66	48.892	111.60
1100	68.93	69.076	138.05
1200	75.19	62.360	137.60
1300	81.46	58.944	140.45
1400	87.72	65.978	153.75
1500	93.99	74.562	168.60

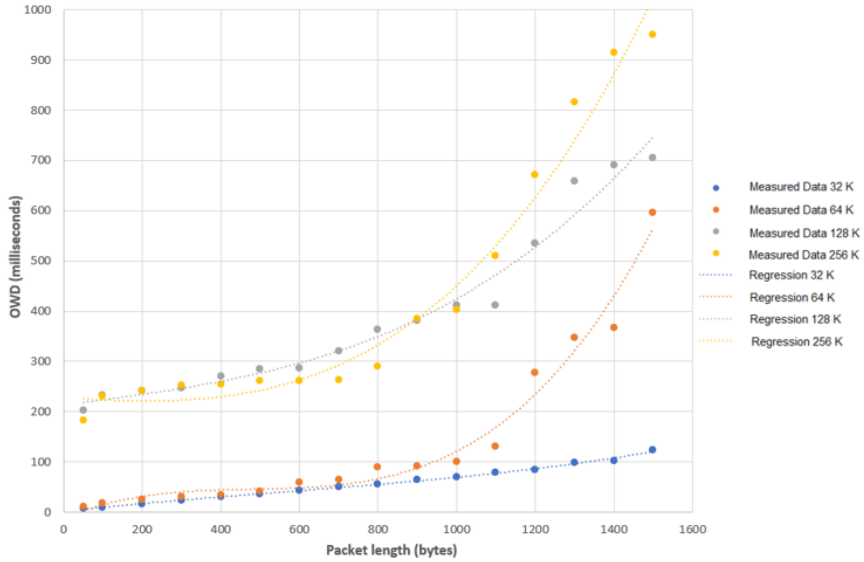


Figure 4. OWD analysis through CBWFQ QoS policy

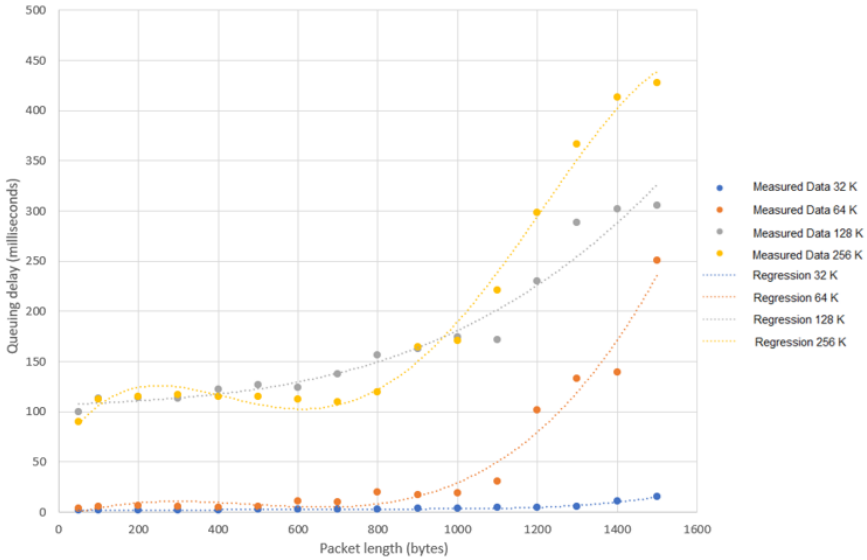


Figure 5. Queuing delay analysis through CBWFQ QoS policy

$$f(x) = \sum_{k=0}^{n} a_k x^k. \quad (3)$$

Table 2. Polynomial coefficients

Delay	Load (Kbps)	a_0	a_1	a_2	a_3	a_4
OWD	32	1.5517	0.087	$-5x10^{-5}$	$3x10^{-8}$	
	64	-13.041	0.3388	$-7x10^{-4}$	$5x10^{-7}$	
	128	213.52	0.1038	$-1x10^{-5}$	$1x10^{-7}$	
	256	227.84	-0.0481	$4x10^{-5}$	$2x10^{-7}$	
Queuing	32	2.517	-0.0115	$-5x10^{-5}$	$-6x10^{-8}$	$2x10^{-11}$
	64	-6.544	0.1381	$-3x10^{-4}$	$2x10^{-7}$	
	128	106.74	0.0205	$-7x10^{-6}$	$6x10^{-8}$	
	256	60.683	0.6351	-0.002	$2x10^{-6}$	$-7x10^{-10}$

3. Methodology: The predictive LPQ model (Third phase)

The LPQ model relies on the predictive behavior of packet lengths to replace or remove the marking mechanism of a DiffServ architecture-based QoS model, which does not mark packet headers but uses the total length field of the packets. It was not considered packet fields to identify protocols and applications for the classification process. The packet length, OWD, and queuing delay are analyzed to model network traffic. The LPQ model employs three queues based on packet length: a small-length packet queue (0 to 200 bytes), a medium-length packet queue (200 to 1300 bytes), and a large-length packet queue (1300 to 1500 bytes). The classifier assigns packets to these queues based on the total length field, using previously estimated traffic models. The Weighted Round Robin (WRR) algorithm is used to set up the scheduler, which takes packets from the software queues and assigns them to the exit interface according to the established bandwidth. This method offers advantages such as ease of implementation, suitability for high-speed networks, and ensuring that each cycle serves all queues to prevent packet depletion in unattended queues. Packets are dispatched more frequently from the queues with the highest weight or priority. Figure 6 presents a scheme of the predictive LPQ model, whereas Table 3 denotes its input parameters.

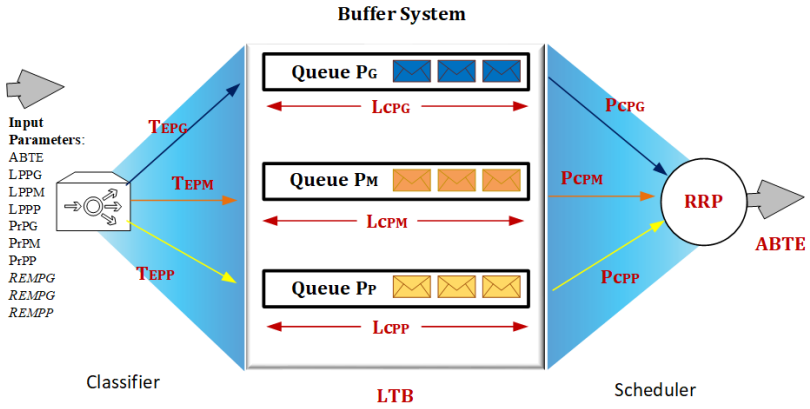


Figure 6. Predictive Length Packet Queuing (LPQ) model

Table 3. Input parameters for the LPQ model

Notation	Description
ABTE	Total link bandwidth
LPPG	Average length of a large packet
LPPM	Average length of a medium packet
LPPP	Average length of a small packet
PrPG	Probability of arrival of a large packet
PrPM	Probability of arrival of a medium packet
PrPP	Probability of arrival of a small packet
REMPG	Medium queuing delay of a large packet
REMPG	Medium queuing delay of a medium packet
REMPP	Medium queuing delay of a medium packet

The LPQ model consists of different input parameters, such as the small, medium, and large packet lengths (LPPP, LPPM, LPPG), the probabilities of occurrence determined by Poisson distributions (PrPP, PrPM, PrPG), and expected mean delay for small, medium and large packets (REMPP, REMPM, REMPG) estimated by polynomial regression. It allows for determining the estimated rate of small, medium, and large packets (TEPP, TEPM, TEPG) arriving in the system. The classification algorithm assigns packets according to their size, that is, in the small, medium, and large packet queues, with lengths defined by the variables LCPG, LCPM, and LCPP.

The queue delay is determined by two variables related to the queue residence delay until the scheduling algorithm sends it to the output interface (RRPP, RRPM, RRPG) and the service time for each packet (RSPP, RSPM, RSPG). This delay depends on the scheduling algorithm (WRR) that takes packets

from the small, medium, and large packet queues based on the assigned weight, i.e., more weight leads to sending more packets from the queue to the output interface. The weights are modified in the simulation scenario to determine how LPQ responds to such behaviors. Finally, queueing utilization levels (UcPP, UcPM, UcPG) control system congestion, whereas packet bursts will determine LPQ behavior in the case of network congestion.

The LPQ model utilizes input parameters from packet length predictive models and queuing delays. This work assumes that packets arrive continuously at the queueing system. IPv4 and TCP traffic models are used as they represent larger traffic volumes and are applicable to the most common applications. Elements that serve as processes, such as UCPG, UCPM, and UCPP provide an idea of the queueing utilization based on the packets assigned in each total expected arrival queue. These queueing utilization levels impact the packet release by the scheduler algorithm. This process will affect the packet queuing residency, which can be declared as residency delay RRPG, RRPM, and RRPP according to the size of the packets. This integrated interaction is useful to lead outcomes related to the Total Queueing System Utilization (UTS). In accordance with QoS and queuing delay theory (Adan & Resing, 2015; Barreiros & Lundqvist, 2015; Shortle, 2018), Table 4 details the system parameter notations, description, and equations used to estimate their values.

Table 4. Parameters denoted for the LPQ model

Notation	Description	Equation
TAE	Total expected arrival rate to the queue	$\mathrm{TAE} = \frac{\mathrm{ABTE}}{8 * \mathrm{LPPM}}$
TEPG	Estimated rate of large packets	$\mathrm{TEPG} = \mathrm{TAE} * \mathrm{PrPG}$
TEPM	Estimated rate of medium packets	$\mathrm{TEPM} = \mathrm{TAE} * \mathrm{PrPM}$
TEPP	Estimated rate of small packets	$\mathrm{TEPP} = \mathrm{TAE} * \mathrm{PrPP}$
LCPG	Large packet queue length	$\mathrm{LCPG} = \frac{\mathrm{LPPG} * \mathrm{TEPG} * REMPG}{\mathrm{TAE}}$
LCPM	Medium packet queue length	$\mathrm{LCPM} = \frac{\mathrm{LPPM} * \mathrm{TEPM} * REMPM}{\mathrm{TAE}}$
LCPP	Small packet queue length	$\mathrm{LCPP} = \frac{\mathrm{LPPP} * \mathrm{TEPP} * REMPP}{\mathrm{TAE}}$
RSPG	Service delay for large packets	$RSPG = \frac{\mathrm{LPPG} * 8}{\mathrm{ABET}}$
RSPM	Service delay for medium packets	$RSPM = \frac{\mathrm{LPPM} * 8}{\mathrm{ABET}}$
RSPP	Service delay for small packets	$RSPP = \frac{\mathrm{LPPP} * 8}{\mathrm{ABET}}$
RS	Average service delay	$RS = \frac{\mathrm{TEPG}}{\mathrm{TAE}} * RSPG + \frac{\mathrm{TEPM}}{\mathrm{TAE}} * RSPM + \frac{\mathrm{TEPP}}{\mathrm{TAE}} * RSPP$
UCPG	Large packet queue utilization level	$\mathrm{UCPG} = \mathrm{TEPG} * RSPG$
UCPM	Medium packet queue utilization level	$\mathrm{UCPM} = \mathrm{TEPM} * RSPM$
UCPP	Small packet queue utilization level	$\mathrm{UCPP} = \mathrm{TEPP} * RSPP$
UTS	Total queuing system utilization	$\mathrm{UTS} = \mathrm{UCPG} + \mathrm{UCPM} + \mathrm{UCPP}$
RRPG	Residence delay for large length packet	$RRPG = RSPG + \frac{(\mathrm{UCPG} * RSPG + \mathrm{UCPM} * RSPM + \mathrm{UCPP} * RSPP)}{(1 - \mathrm{UCPP} - \mathrm{UCPM})}$
RRPM	Residence delay for medium length packet	$RRPM = RSPM + \frac{(\mathrm{UCPG} * RSPG + \mathrm{UCPM} * RSPM + \mathrm{UCPP} * RSPP)}{(1 - \mathrm{UCPM})}$
RRPP	Residence delay for small length packet	$RRPP = RSPP + \frac{(\mathrm{UCPG} * RSPG + \mathrm{UCPM} * RSPM + \mathrm{UCPP} * RSPP)}{(1 - \mathrm{UCPP})}$
RR	Average residence delay	$RR = \frac{\mathrm{TEPG}}{\mathrm{TAE}} * RRPG + \frac{\mathrm{TEPM}}{\mathrm{TAE}} * RRPM + \frac{\mathrm{TEPP}}{\mathrm{TAE}} * RRPP$

In Table 4, the parameters that comprise the LPQ model are detailed, starting with TAE, which defines the maximum number of packets that can be received simultaneously on an input interface, depending on the available bandwidth. The variables TEPG, TEPM, and TEPP represent the expected rates of large, medium, and small packets, respectively, and are determined by the input parameters corresponding to their respective arrival probabilities, denoted as PrPG, PrPM, and PrPP. The packet queue length parameters LCPG, LCPM, and LCPP establish the queue length for large, medium, and small packets, based on the expected packet arrival rates and their respective average lengths. Similarly, RSPG, RSPM, and RSPP quantify the service delay associated with each packet type (large, medium, or small), based on their average lengths. In addition, UcPG, UcPM, and UcPP measure the system's utilization level based on the estimated rate of each packet size. The sum of these three parameters represents the total queuing utilization of the system. Lastly, RRPG, RRPM, and RRPP capture the packet residence delay relative to utilization, considering the delays associated with the respective packet queues.

4. Simulation and results

It was used MATLAB Simulink for modeling the LPQ with an ABTE value of 256 Kbps. This allowed us to compare the delayed values of the simulated model with those obtained in an experimental scenario using the CBWFQ QoS policy. The estimated delay values were used to size the queue length. The network traffic model estimates calculated in Espinal *et al.* (2019a, 2019b, 2019c, 2020) provided the packet lengths and their probabilities. The LPQ model performance was simulated by varying the probability of packet length occurrence through a traffic model, where the occurrence of large packets was higher than small packets, and vice versa. The model's behavior can be analyzed across different scenarios of packet length, either minimum or maximum. In all scenarios, the parameter *large packet length* ranged from 100 to 1500 bytes, while *small packet length* was set to 80 bytes.

In the first scenario, the occurrence probabilities for the packets were PrPG = 0.5005, PrPM = 0.0686, and PrPP = 0.4309, and the utilization of the queuing system ranged from 0.10 to 1.07. In the second scenario, priorities were adjusted for small and large packet lengths, wherein the network traffic model assigned a higher probability to large packets (PrPG = 0.70) compared to small packets (PrPP = 0.23). The queuing system utilization ranged from 0.09 to 1.09 under moderate congestion. For the third scenario, the traffic model was configured with a higher probability for small packets (PrPP = 0.70) compared to large packets (PrPG = 0.23). The queuing system utilization ranged from 0.17 to 1 under moderate congestion, where above 1 represents packet congestion, loss, or storage in buffers.

Due to Poisson distribution considerations, the distribution fell in a bimodal behavior, where the packets that experienced considerable changes were either small or large. Figure 7 presents the results for the queuing delay obtained from LPQ simulations with different occurrence probabilities of large packets (PrPG = 0.70, 0.50, 0.23). Figure 8 compares the queuing delay results by introducing a variation in the LPPP to observe how the delay values change. Figure 9a illustrates the queuing delay behavior for PrPG = 0.70 and PrPP = 0.23, while Figure 9b shows the results based on the modification of probabilities PrPG = 0.23 and PrPP = 0.70.

Based on the information collected, it was estimated a predictive model for queuing delay in the LPQ model using polynomial regression. Table 5 depicts the statistical analysis for the three scenarios, listing the respective estimated function model and conditions. The calculated standard error indicates high precision of the predictive model, and the *p*-value confirms the reliability of the results.

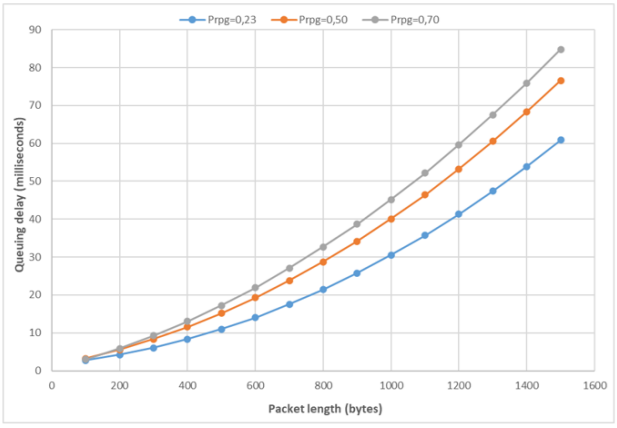


Figure 7. Queuing delay analysis using LPQ for large packets

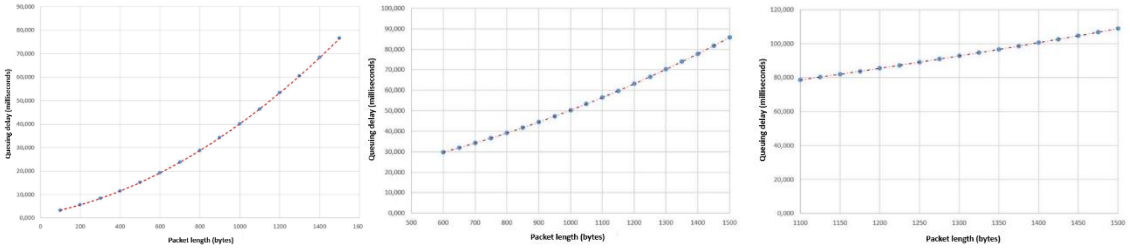
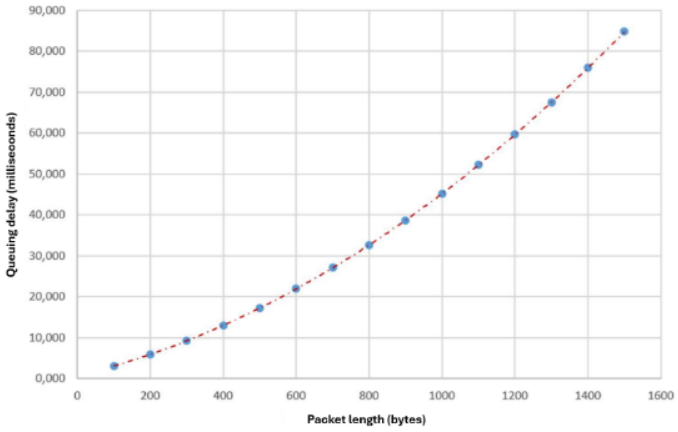
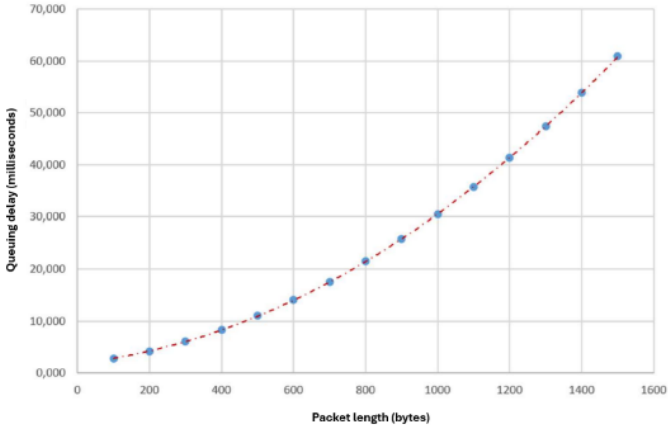


Figure 8. Queuing delay comparison in the first scenario with LPPP = 80, 500, and 1,000 bytes



a) Probabilities of PrPG=0.70 and PrPP=0.23



b) Probabilities of PrPG = 0.23 and PrPP = 0.70

Figure 9. Queuing delay comparison for a) second scenario and b) third scenario

Figure 10 compares the queueing delay values collected from the LPQ model simulation with those obtained during the experimental setting using CBWFQ. The conditions were similar concerning bandwidth and traffic load. LPQ demonstrates lower queuing delays compared to CBWFQ. The queuing system utilization values ranged from 0.10 to 1.07, indicating that under moderate congestion the LPQ model achieves lower queuing delays. The hypothesis test compares the means of LPQ and CBWFQ, stated as $H_0: \mu_1 = \mu_2$ versus $H_1: \mu_1 \neq \mu_2$. It was assessed the queuing delay values estimated by the LPQ model against experimental data obtained using the CBWFQ QoS policy. As shown in Table 6, the resulting p-value obtained is less than 0.05, leading to the rejection of the null hypothesis and providing clear statistical evidence that the means differ significantly.

Table 5. LPQ model statistical analysis for the different scenarios with the estimated function model

Scenario	Model	Bytes	Estimate values	Standard Error	*p*-value
1	Estimated model $f(x) = 2.27x10^{-5}X^2 + 0.0159X + 1.5090$				
	(Intercept)	LPPP = 80	1.5090	0.0432	$1.9137x10^{-13}$
	x_1		0.0159	0.0001	$3.5431x10^{-20}$
	x_2		$2.2762x10^{-5}$	$7.5446x10^{-8}$	$1.1837x10^{-24}$
	Estimated model $f(x) = 2.21x10^{-5}X^2 + 0.0157X + 12.3808$				
	(Intercept)	LPPP = 500	12.3808	0.0349	$1.3150x10^{-32}$
	x_1		0.0157	$6.95x10^{-5}$	$1.8646x10^{-29}$
	x_2		$2.2181x10^{-5}$	$3.2855x10^{-8}$	$4.5223x10^{-37}$
	Estimated model $f(x) = 2.30x10^{-5}X^2 + 0.0156X + 33.5426$				
	(Intercept)	LPPP = 1,000	33.5426	0.0338	$2.4697x10^{-35}$
	x_1		0.0156	$5.23x10^{-5}$	$4.8903x10^{-28}$
	x_2		$2.3028x10^{-5}$	$2.0114x10^{-8}$	$3.3207x10^{-36}$

Continuation of Table 5

Scenario	Model	Bytes	Estimate values	Standard Error	*p*-value
2	Estimated model $f(x) = 2.29x10^{-5}X^2 + 0.0216X + 0.6799$				
	(Intercept)	PrPG = 0.70 PrPP = 0.23	0.6799	0.0361	$2.7727x10^{-10}$
	x_1		0.0216	$1.04x10^{-4}$	$1.0035x10^{-22}$
	x_2		$2.2962x10^{-5}$	$6.3033x10^{-8}$	$1.2325x10^{-25}$
3	Estimated model $f(x) = 2.11x10^{-5}X^2 + 0.0075X + 1.8806$				
	(Intercept)	PrPG = 0.23 PrPP = 0.70	1.8806	0.0416	$8.8258x10^{-15}$
	x_1		0.0075	$1.20x10^{-4}$	$1.7121x10^{-16}$
	x_2		$2.1165x10^{-5}$	$7.2629x10^{-8}$	$1.7948x10^{-24}$

5. Conclusions

This paper proposed a predictive QoS model based on packet length analysis to reduce queuing delays in intermediate devices. The LPQ model is well-suited for network traffic generated by ordinary applications on user devices within hybrid networks. By applying queuing and QoS theory, it was derived expressions to estimate queuing delay, while computer simulations were conducted to evaluate the accuracy of these numerical results. The LPQ model demonstrated lower delay values compared to benchmark models. With this QoS policy, data network providers can predict queuing delays more effectively and enhance service quality for applications based on packet length. The simulations confirmed that LPQ achieves reduced delays for real applications.

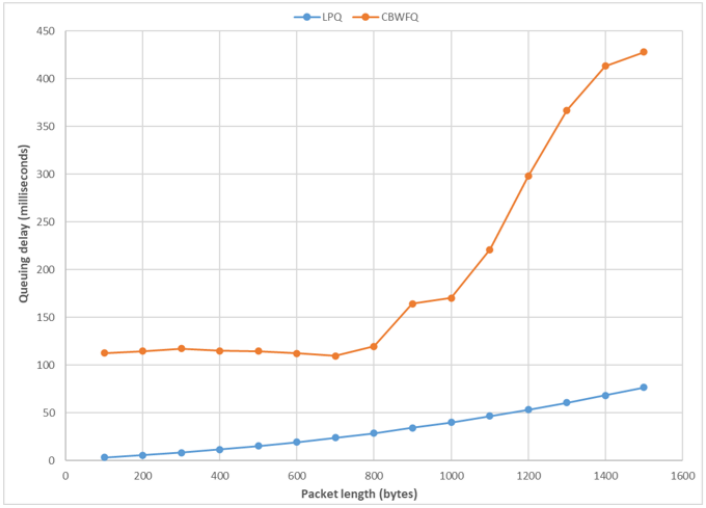


Figure 10. LPQ model and CBWFQ QoS policy comparison

Table 6. ANOVA between LPQ and CBWFQ

Situation	Sum of squares	Degrees of Freedom	Quadratic mean	F	*p*-value
Between groups	388772.658	1	388772.658	58.955	<0.05
Within groups	382477.255	58	6594.435		
Total	771249.913	59			

Some limitations can be noticed in the present proposal. While the LPQ model evaluates packet lengths ranging from 50 to 1500 bytes, the impact of very large or very small packets, or highly irregular packet sizes, is not fully explored. Different variations in packet length might affect queuing delay performance. In addition, the simulations focus on point-to-point topologies, where the performance of the LPQ model would become more complex in other topologies that involve multiple hops, different types of connections, or heterogeneous network environments, to mention a few. Therefore, future work could focus on extending this model to other network types and exploring its application through other QoS conditions in broader areas that may include network security and traffic engineering.

References

1. Adan, I., & Resing, J. (2015) *Queueing systems*. Eindhoven University of Technology. https://iadan.win.tue.nl/queueing.pdf.
2. Alaya, B., Khan, R., Moulahi, T., & Khediri, S. E. (2021) Study on QoS management for video streaming in vehicular Ad Hoc Network (VANET). *Wireless Personal Communications*, 118(4), 2175–2207. DOI:10.1007/s11277-021-08118-7.
3. Barreiros, M., & Lundqvist, P. (2015) *QoS-enabled networks: Tools and foundations*. New Jersey: John Wiley & Sons.
4. Botta, A., Dainotti, A., & Pescapé, A. (2012) A tool for the generation of realistic network workload for emerging networking scenarios. *Computer Networks*, 56(15), 3531–3547. DOI:10.1016/j.comnet.2012.02.019.
5. Chefrour, D. (2021) One-Way delay measurement from traditional networks to SDN: A survey. *ACM Computing Surveys*, 54(7), 1–35. DOI:10.1145/3466167.
6. Csoma, A., Toka, L., & Gulyas, A. (2015) On lower estimating internet queuing delay. In: *Proceedings of the 38th International Conference on Telecommunications and Signal Processing (TSP)*, Prague, July 2015. IEEE, 299–303. DOI:10.1109/TSP.2015.7296272.
7. Daza Alava, Y., Zambrano, D. M., Cedeño Palma, E., Chancay Garcia, L., & Cruz Felipe, M. (2021) Evaluation of quality of service in VoIP traffic using the E model. In T. Guarda, F. Portela, & M. F. Santos (Eds.), *Advanced Research in Technologies, Information, Innovation and Sustainability*, 1485, 34–43. Springer International Publishing. DOI:10.1007/978-3-030-90241-4_3.
8. Di Mauro, M., & Liotta, A. (2020) An experimental evaluation and characterization of VoIP over an LTE-A Network. *IEEE Transactions on Network and Service Management*, 17(3), 1626–1639. DOI:10.1109/TNSM.2020.2995505.
9. Espinal, A., Estrada, R., & Monsalve, C. (2019a) Modelling TCP/IP traffic of a convergent campus wireless network. *International Journal of Circuits, Systems and Signal Processing*, 13, 611–616.
10. Espinal, A., Estrada, R., & Monsalve, C. (2019b) Traffic analysis of internet applications on mobile devices over LTE and Wireless networks. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, E22, 81–94.
11. Espinal, A., Estrada, R., & Monsalve, C. (2019c) Traffic model using a novel sniffer that ensures the user data privacy. *MATEC Web of Conferences*, 292, 03002. DOI:10.1051/matecconf/201929203002.
12. Espinal, A., Estrada, R., & Monsalve, C. (2020) Benchmarking and modelling of IP traffic in a heterogeneous campus network (Análisis comparativo y modelamiento del tráfico IP en una red de campus heterogénea). *Investigación Operacional*, 41(4), 494–504 (in Spanish).
13. Espinal, A., Estrada, R., Monsalve, C., Solorzano, M., & Muñoz, A. (2024) Measurement and modelling of unidirectional delay in a point-to-point link (Medición y modelamiento del retardo unidireccional en un enlace punto a punto). *Investigación Operacional*, 44(3), 395–405 (in Spanish).
14. Ferencz, B., & Kovacshazy, T. (2014) One-way delay measurement system for local area network delay and jitter characterization. In: *Proceedings of the 15th International Carpathian Control Conference (ICCC)*, Velke Karlovice, May 2014. IEEE, 14–18. DOI:10.1109/CarpathianCC.2014.6843561.
15. Fiedler, M., Hossfeld, T., & Tran-Gia, P. (2010) A generic quantitative relationship between quality of experience and quality of service. *IEEE Network*, 24(2), 36–41. DOI:10.1109/MNET.2010.5430142.
16. Floyd, S. (2008) *RFC 5166: Metrics for the evaluation of congestion control mechanisms*. RFC Editor. DOI:10.17487/rfc5166.
17. Ghazel, C., & Saïdane, L. (2015) Satisfying QoS requirements in NGN networks using a dynamic adaptive queuing delay control method. *Procedia Computer Science*, 56, 225–232. DOI: 10.1016/j.procs.2015.07.203.
18. Gheisari, M., Alzubi, J., Zhang, X., Kose, U., & Saucedo, J. A. M. (2020) A new algorithm for optimization of quality of service in peer to peer wireless mesh networks. *Wireless Networks*, 26(7), 4965–4973. DOI:10.1007/s11276-019-01982-z.
19. Gómez, G., Pérez, Q., Lorca, J., & García, R. (2014) Quality of service drivers in LTE and LTE-A networks. *Wireless Personal Communications*, 75(2), 1079–1097. DOI:10.1007/s11277-013-1409-0.
20. Hodroj, A., Ibrahim, M., & Hadjadj-Aoul, Y. (2021) A survey on video streaming in multipath and multihomed overlay networks. *IEEE Access*, 9, 66816–66828. DOI:10.1109/ACCESS.2021.3076464.
21. Kempa, W. M. (2013) A direct approach to transient queue-size distribution in a finite-buffer queue with AQM. *Applied Mathematics & Information Sciences*, 7(3), 909–915. DOI:10.12785/amis/070308.

22. Kompella, R. R., Levchenko, K., Snoeren, A. C., & Varghese, G. (2012) Router support for fine-grained latency measurements. *IEEE/ACM Transactions on Networking*, 20(3), 811–824. DOI:10.1109/TNET.2012.2188905.
23. Lindeberg, M., Kristiansen, S., Plagemann, T., & Goebel, V. (2011) Challenges and techniques for video streaming over mobile ad hoc networks. *Multimedia Systems*, 17(1), 51–82. DOI:10.1007/s00530-010-0187-8.
24. Liu, J. (2014) A novel method for estimating the variable and constant components of one-way delays without using the synchronized clocks. In: *Proceedings of International Conference on Computing, Networking and Communications (ICNC)*, Honolulu, February 2014. IEEE, 1028–1033. DOI:10.1109/ICCNC.2014.6785479
25. Narayanaswamy, S. & Rajan, Sh. J. (2021) Impact of queuing disciplines on the performance of multi-class traffic in a network. *Information Technology in Industry*, 9(1), 691–697. DOI:10.17762/itii.v9i1.189.
26. Salehin, K., Rojas-Cessa, R., & Kwon, K. W. (2019) COMPRESS: A self-sufficient scheme for measuring queueing delay on the Internet routers. In: *Proceedings of International Conference on Computing, Networking and Communications (ICNC)*, Honolulu, February 2019. IEEE, 624–629. DOI:10.1109/ICCNC.2019.8685606.
27. Shortle, J. F. (2018) *Fundamentals of queueing theory.* John Wiley & Sons.
28. Sukhov, A. M., Kuznetsova, N. Y., Pervitsky, A. K., & Galtsev, A. A. (2016) Generating function for network delay. *Journal of High Speed Networks*, 22(4), 321–333. DOI:10.3233/JHS-160552.
29. Sumarsono, A., & Rodriguez, M. (2021) Speculative packet dispatch for virtual output queuing architecture using LSTM recurrent neural network. *International Journal of Information and Communication Sciences*, 6(2), 38–45. DOI: 10.11648/j.ijics.20210602.13.
30. Ulbricht, M., & Wagner, J. (2016) Accelerated processing delay optimization in hierarchical networks using low cost hardware. In: *Proceedings of the 10th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Prague, July 2016. IEEE, 1–6. DOI:10.1109/CSNDSP.2016.7573903.
31. Vassilakis, V. G., Moscholios, I. D., & Logothetis, M. D. (2018) Quality of service differentiation in heterogeneous CDMA networks: A mathematical modelling approach. *Wireless Networks*, 24(4), 1279–1295. DOI:10.1007/s11276-016-1411-z.
32. Wenbin, Y., Yin, C., Ming, Z., & Dongbin, W. (2017) QoS-oriented packet scheduling scheme for opportunistic networks. *The Journal of China Universities of Posts and Telecommunications*, 24(3), 51–57. DOI:10.1016/S1005-8885(17)60211-5.
33. Yihunie, F., & Abdelfattah, E. (2018) Simulation and analysis of Quality of Service (QoS) of voice over IP (VoIP) through local area networks. In: *Proceedings of the 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, New York, November 2018. IEEE, 598–602. DOI:10.1109/UEMCON.2018.8796802.
34. Zakariyya, I., & Rahman, M. N. (2015) Bandwidth guarantee using class based weighted fair queue (CBWFQ) scheduling algorithm. *International Journal of Digital Information and Wireless Communications*, 5(3), 152–157. DOI:10.17781/P001675.