Scientific Research Publishing

Open Journal of Applied Sciences, 2025, 15(9), 2621-2637
https://www.scirp.org/journal/ojapps
ISSN Online: 2165-3925
ISSN Print: 2165-3917

Determination of the Upper Limit of the Delay in the XG-PON Network

Akanza Konan Ricky N'dri[1], Nouho Ouattara[1], Jean Edgard Gnimassoun[2]

[1]Département Mathématiques et Informatique, Université Alassane Ouattara, Bouaké, Côte d'Ivoire
[2]Département Informatique et Analyse de Données, Université de San Pedro, San Pedro, Côte d'Ivoire
Email: ricky.akanza@uao.edu.ci, nouho_ouattara@yahoo.fr, gnimjean@gmail.com

How to cite this paper: N'dri, A.K.R., Ouattara, N. and Gnimassoun, J.E. (2025) Determination of the Upper Limit of the Delay in the XG-PON Network. *Open Journal of Applied Sciences*, **15**, 2621-2637.
https://doi.org/10.4236/ojapps.2025.159176

Received: August 1, 2025
Accepted: September 9, 2025
Published: September 12, 2025

Abstract

One of the optimization solutions in XG-PON networks is the proposal of an efficient bandwidth allocation algorithm that impacts other Quality of Service (QoS) parameters such as delay, jitter, and packet loss. However, a better allocation algorithm alone is not sufficient to guarantee delay, and maintaining it within a certain limit remains a challenge. In XG-PON networks, delay is defined as the time a packet spends from its arrival in the Optical Network Unit (ONU) buffer until its transmission to the Optical Line Terminal (OLT). It requires special attention, especially since this network offers delay-sensitive services such as videoconferencing, telemedicine, and IPTV. This article aims to determine the upper bound of the delay, *i.e.*, a delay that cannot be exceeded by packets as long as the network does not experience failures. By creating our dataset using the simulation log file of the XGPON network module, we calculate certain parameters, such as the inter-arrival time of packets, the arrival rate in the ONU buffer, the service rate, the bandwidth service time by the OLT, and the transmission time (departures) of packets from the ONU to the OLT. From these parameters, we propose a delay approach by linking cumulative arrivals $A(t)$, cumulative services $S(t)$, and cumulative departures $D(t)$. Using a hypothesis test to find the critical rejection region, we determine the upper bound of the delay that packets should not exceed. This limit is set at 3 ms and is validated by exact queuing theory models.

Keywords

Delay Approach, Hypothesis Test, Normal Distribution, Upper Bound of Delay

1. Introduction

All real-time applications (videoconferencing, telemedicine, online gaming, etc.)

require a certain level of Quality of Service (QoS) for smooth operation. This QoS is expressed in terms of delay, jitter, throughput, and packet loss. For such traffic, end-to-end delay and jitter are influential performance metrics. These metrics determine the amount of distortion that occurs between a source and a destination through a packet flow between them [1].

In XG-PON networks, one of the main challenges is proposing an efficient bandwidth allocation algorithm. The performance of this allocation must impact other QoS parameters, such as delay, jitter, and packet obsolescence, as demonstrated by the authors in [2]-[4].

An acceptable end-to-end delay is closely tied to Quality of Service (QoS), which is why, according to N. Gore *et al.* in [5], special attention is needed to ensure packet transmission and real-time application performance.

According to the authors in [6], the delay of less than 1.5 ms set by the International Telecommunication Union (ITU) for the XG-PON network cannot always be respected. Indeed, when the network includes a certain number of Optical Network Units (ONUs), running programs, and applications with variable priorities, maintaining this standard becomes challenging.

What would be the delay limit to ensure packet transmission in the XG-PON network?

This article presents a statistical and probabilistic approach to determining the delay limit. This limit can serve as a means for operators to test the performance of XG-PON.

The article is organized as follows: related work is presented in Section 2. Section 3 discusses calculation methods, approaches for determining the delay, and its limit. Section 4 focuses on performance evaluation, followed by the analysis in Section 5. We conclude with Section 6.

2. Related Works

The limitation of delay and jitter has been widely used as a performance metric in various types of networks, especially for real-time traffic. G. Geleji *et al.* in [7] report an analysis of jitter in a flow, defining it as the percentile of the time between the arrivals of successive packets at the destination. Unfortunately, their solution for jitter was obtained numerically, introducing an error in the calculation. In [8], the authors H. Dbira *et al.* propose an approach for estimating jitter and delay for periodic traffic with Poisson arrivals and non-Poisson first-come, first-served queues. This approach is based on steady-state analysis methods, which do not provide significant characteristics of jitter and delay, such as probability. Unlike steady-state approaches, Thombre *et al.* in [9] presented an analytical expression for jitter and end-to-end delay using transient queuing analysis, similar to N. Fai *et al.* in [10], who used transient analysis methods to obtain the queue length over a random time interval. These analytical expression-based approaches are applied to lightly configured networks.

Lübben *et al.*, in [11], explore stochastic network calculus to estimate the avail-

able bandwidth of wireless communication systems. They consider the network as a time-invariant deterministic system where traffic rates and delays are governed by an unknown factor. As mentioned by the authors in [5], deterministic performance bounds are pessimistic and often overestimate network resources.

Guang Yang *et al.* in [12] use network calculus to estimate the end-to-end delay cost in millimeter-wave and multi-hop (mmWave) networks. They obtain the upper limit of the service curve of mmWave networks with the moment-generating function (MGF), but the delay analysis for vehicle-to-vehicle (V2V) and vehicle edge computing (VEC) using this technology was not considered.

Azuaje O. *et al.* in [13] use stochastic network calculus in end-to-end delay analysis to address the issue of stochastic dependency between traffic flows of different multi-hop wireless sensor network (WSN) nodes. Miao *et al.* use an analytical model based on stochastic network calculus (SNC) to quantitatively study end-to-end performance limits in network function virtualization in [14]. However, Wang *et al.* in [15] argue that all previous models for calculating the delay limit generally rely on simple assumptions based on architecture properties that lack precision. According to Bondorf in [16], the fastest available analysis of FIFO network calculus suffers from limitations with outlier bounds. They demonstrate that a feature called Flow Prolongation (FP) significantly improves the accuracy of delay bounds. Unfortunately, FP must be run very frequently in FIFO network calculus analysis, and each time, it creates a set of alternative networks with exponentially growing prolongations. FP is therefore not scalable and is beyond reach for exhaustive analysis of large networks. Thus, Geyer *et al.* [17] introduce DeepFP, an approach to scaling FP by predicting prolongations using machine learning.

Besides the use of stochastic network calculus, Megha S. *et al.*, in [18], use a decision tree to design an end-to-end traffic adaptation algorithm on LTE interfaces to achieve an appropriate level of jitter for better delay. Although they find a value of 5 ms, they are still working to limit it across all interfaces. Gore *et al.*, in [5], introduce a probabilistic network calculus approach that uses moment-generating functions to analyze real-time traffic delay and jitter and keep them within bounds to ensure a certain level of QoS.

Y. Mansour *et al.* in [19], Study jitter control in QoS-guaranteed networks, via online competitive analysis and algorithms that regulate jitter.

H. Toral-Cruz *et al.* in [1] define it as an influential performance metric to determine the amount of distortion that occurs between a source and a destination via a packet flow.

Although jitter is a performance indicator, it can be measured in different ways. For Y. Mansour et al., jitter can be explored using two concepts: delay jitter (total differences in delays between packets) and rate jitter (inter-arrival differences/interval between packets). Elsewhere, H. Dbira *et al.*, in [20], measure jitter as the variation in inter-packet delay, *i.e.*, the difference between the arrival times of packets. The authors in [21] and [22] take the delay variation of consecutive pack-

ets as the value of jitter, measuring it as the difference between the current packet's delay and the previous one.

3. NS3 Configuration

For the NS-3 configuration, we used the following experimental conditions:

The network is based on an XG-PON configuration composed of a central Optical Line Terminal (OLT) and 16 Optical Network Units (ONUs).

Traffic generation involves variable packet arrival rates. A Poisson arrival process is used for packet generation at the ONUs, following exponential distribution assumptions for inter-arrival times. The analysis revolves around a cycle time of 125 μs, which allows for deriving shorter delays for the simulations.

The proposed approach sets the mean delay to 1.5 ms as a threshold value, based on ITU standards, for acceptable delays in real-time applications.

One OLT and 16 ONUs are connected via point-to-multipoint links and configured to use UDP as the traffic model, with a simple client-server architecture as shown in Figure 1.

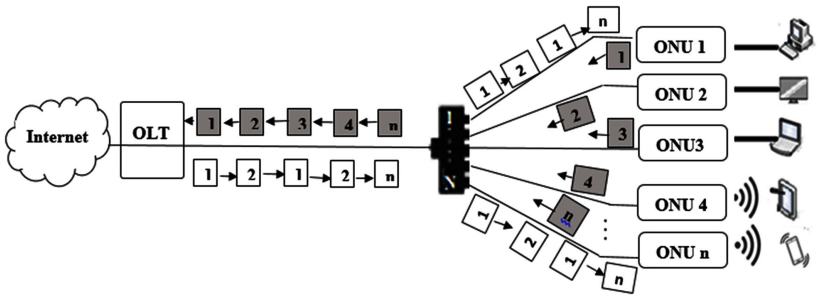

Figure 1. Simulation network architecture.

The service rate is set to 8 Mbps and the simulation duration to 15 seconds.

The traffic load (tasks to be executed on each ONU) varies from 80.106 Mbps, 90.106 Mbps, 100.106 Mbps, 110.106 Mbps, 120.106 Mbps, 130.106 Mbps, 140.106 Mbps and 150.106 Mbps.

NS3 Configuration Algorithm

Data and default parameters
numONUs ← 16
simTime ← 15 seconds
serviceRate ← 50e6 (50 Mbps)
Read command-line arguments
Option --numONUs to override numONUs
Global configuration
Enable logs
Enable LOG_LEVEL_INFO for "Simulation_XG_PON"
Topology creation
Create nodes

```
olt ← NodeContainer(1)
onus ← NodeContainer(numONUs)
olt.Create(1)
onus.Create(numONUs)
Configure Point-to-Multipoint links
p2mp ← PointToPointHelper
p2mp.SetDeviceAttribute("DataRate", "1Mbps")
p2mp.SetChannelAttribute("Delay", "2ms")
Install devices and network stack
Store OLT-ONU link devices
onuDevices ← empty vector of NetDeviceContainer
For i from 0 to numONUs-1
  onuDevice ← p2mp.Install(olt.Get(0), onus.Get(i))
  onuDevices.push_back(onuDevice)
Install the Internet stack
InternetStackHelper internet
internet.Install(olt)
internet.Install(onus)
IP address assignment
Configure IPv4
ipv4 ← Ipv4AddressHelper
ipv4.SetBase("10.1.1.0", "255.255.255.0")
Assign addresses
For i from 0 to numONUs-1
  ipv4.Assign(onuDevices[i])
UDP applications configuration (traffic)
Define UDP port
port ← 9
UDP Echo Server
echoServer ← UdpEchoServerHelper(port)
serverApps ← echoServer.Install(onus)
serverApps.Start(Seconds(1.0))
serverApps.Stop(simTime)
UDP Echo Client
echoClient ← UdpEchoClientHelper(ipv4.NewAddress(), port)
echoClient.SetAttribute("MaxPackets", UintegerValue(1000))
echoClient.SetAttribute("Interval", TimeValue(Seconds(1.0)))
echoClient.SetAttribute("PacketSize", UintegerValue(1024))
Install and deploy clients
Deploy clients on the ONUs
clientApps ← empty ApplicationContainer
For i from 0 to numONUs-1
  echoClient.SetRemote(ipv4.NewAddress(), port)
```

```
    clientApps.Add(echoClient.Install(onus.Get(i)))
Start clients
clientApps.Start(Seconds(2.0))
clientApps.Stop(simTime)
Tracing and logging
ASCII tracing
ascii ← AsciiTraceHelper
p2mp.EnableAsciiAll(ascii.CreateFileStream("simulation_xg_pon.tr"))
PCAP tracing
p2mp.EnablePcapAll("simulation_xg_pon")
Run the simulation
Schedule stop
Simulator::Stop(simTime)
Run the simulation
Simulator::Run()
Simulator::Destroy()
End of program
Return 0
```

4. Calculation of Queueing Parameters

To determine the upper bound of the delay that must not be exceeded by packets while the network does not experience failures, we use the simulation log file from the xg-pon module. From this file, we compute certain parameters such as the arrival instants of packets in the buffers of the Optical Network Units (ONUs), the service times of the bandwidth by the Optical Line Terminal (OLT), as well as the departure times of packets from the ONU to the OLT. From these parameters, we derive the delay approach in Equation (5) of section 4.1 and the upper bound of the delay with the model in Equation (7) of section 4.2.

Determination of the Arrival Rate (λ) and Inter-Arrival Time

This parameter represents the average number of packets arriving per unit of time. To calculate it, we apply Little's theorem [23], which defines the arrival rate as follows:

$$\lambda = N/T \tag{1}$$

where N is the average number of packets in the system, λ is the arrival rate, and T is the average time a packet spends in the system, specifically $T = 8 \times 125$ µs.

Inter-Arrival Time of Packets

According to [24], the inter-arrival time of packets in the ONU queue follows an exponential distribution $\varepsilon(\lambda)$ with the cumulative distribution function given by:

$$A(t) = 1 - e^{-\lambda t} \tag{2}$$

Determination of the Service Rate (μ)

The service rate is the average number of services that the server can perform

per unit of time. In the XG-PON network, the service rate represents the average bandwidth to be allocated by the OLT in an XG-PON cycle (125 μs). According to queuing theory in [25], the service rate is given by:

$$\mu = 1/TS \tag{3}$$

Average Service Time

According to the same authors in [25], the average service time also follows an exponential distribution $\varepsilon(\mu)$ with the cumulative distribution function given by:

$$S(t) = 1 - e^{-\mu t} \tag{4}$$

4.1. Delay Limitation Approach

These parameters allow us to propose a delay approach. With the help of a normal distribution and hypothesis tests, we can define the most unfavorable upper delay limit in the XG-PON network.

To determine the delay function, a connection is made between cumulative arrivals $A(t)$, cumulative services $S(t)$, and cumulative departures $D(t)$ of the packets, as shown in the figure below.

In this figure, at stage 1, all packet arrivals in the buffer of the ONU during the period (s,t) are accumulated in $A(s,t)$. Once in the ONU's waiting queue, and at the request of the OLT, a report on the queue size and the priority types of the packets is sent to it, as mentioned in stage 2 in Figure 2. After analysis and processing, the OLT, in a cumulative service $S(s,t)$ during a given period (s,t), allocates the bandwidth based on available resources and priorities. It also defines the start time for transferring the packets, as represented in stage 3. Thus, at the indicated departure time $D(s,t)$, the ONU will be able to transfer its packets, as mentioned at stage 4.

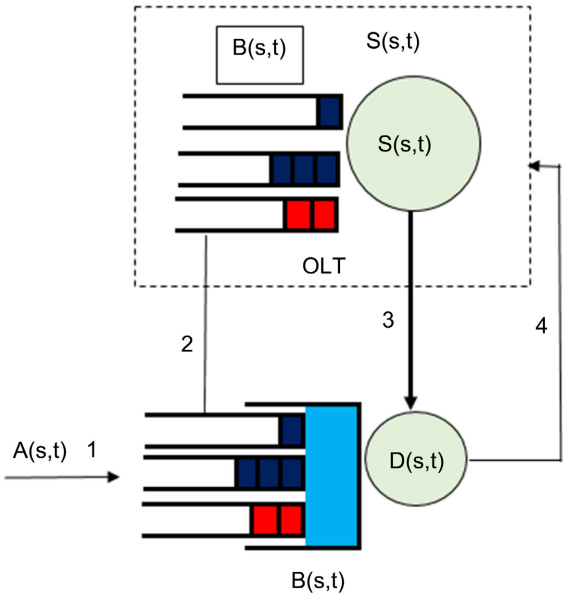

Figure 2. ONU's buffer.

According to the packet transmission process described in Figure 2 above, the arrival time of a packet is always less than that of the service. These two times also precede that of the transfer or departure of the packets, which is set by the OLT. We thus propose the following delay approach:

$$W(t) = \min\left\{v \geq 0, \min_{\tau\in[0,\tau]}\left\{A(\tau) + S(\tau,t)\right\} \leq D(t+v)\right\} \quad (5)$$

Where is the delay between the arrival and departure of the packet, $A(\tau)$ is the cumulative arrival of packets at a given time *t*, *S*(*t*) is the bandwidth allocated to packets at a given time *t*, *D*(*t*) is the departure or transmission of the packet at a given time *t*.

4.2. Determination of the Upper Limit on Delay

We define an operational and scientific context for evaluating latency in an XG-PON network dedicated to virtual reality and cloud gaming applications. The objective is to ensure an end-to-end round-trip latency of less than or equal to 3 ms by combining an XG-PON access architecture with appropriate QoS (quality of service) mechanisms, while providing a reproducible experimental plan. The document distinguishes two analysis frameworks: models from exact queueing theory and an analytical model, used as a fast baseline, based on realistic simulation data (NS-3) that incorporates upstream medium sharing, TDMA scheduling, and network overheads. The objective of this article is to determine the upper bound of the delay, which constitutes the limit that packets should not exceed in the queue. To achieve this, we propose to find the critical rejection region in this study by performing the test to reject H0: $d \leq d_0$ (normal) against H1: $d > d_0$ (rejection).

Let $\bar{W}$, the average delay of our sample (dataset) of size $n > 30$, under H0 ($d = d_0$), the central limit theorem implies that $U = \dfrac{\bar{W} - d_0}{\delta/\sqrt{n}}$ converges in law towards normal law centered reduced because they are independent and identically distributed then $E\left[W_i^2\right] \leq +\infty$.

If ε the violation error is the risk of the first kind then the critical region of rejecting H0 is defined by $P\left[\bar{W} > d\right] = \varepsilon$.

$$P\left[\bar{W} > d\right] = P\left[\frac{\bar{W} - d_0}{\delta/\sqrt{n}} > \frac{d - d_0}{\delta/\sqrt{n}}\right] = P\left[U > \frac{d - d_0}{\delta/\sqrt{n}}\right] = \varepsilon \quad (6)$$

$$\frac{d - d_0}{\delta/\sqrt{n}} = \varepsilon \rightarrow d = \varepsilon \times \frac{\delta}{\sqrt{n}} + d_0 \quad (7)$$

where *d* denote the limit of the delay, d_0 denote the delay threshold (1.5 ms); *n*, the size of our sample. $\bar{W}$ denote the average delay of our sample δ denote the standard deviation of the delays and ε, violation error and first type risk.

5. Performance Evaluation

We calculate the various parameters determined in section 4.3.1 above using the log file from the simulation of the XG-PON module in NS3. This simulation was

conducted on 16 ONUs, varying the traffic load (tasks to be executed on each ONU) from 80 × 10^6 Mbps, 90 × 10^6 Mbps, 100 × 10^6 Mbps, 110 × 10^6 Mbps, 120 × 10^6 Mbps, 130 × 10^6 Mbps, 140 × 10^6 Mbps and 150 × 10^6 Mbps. From these parameters, we derive the delay approach in Equation (5) from section 3.1 and the upper delay limit using the model in Equation (7) from section 4.2.

For the validation of the models determining the upper delay limit, Wang Miao *et al.* [14] suggest that they be compared to those from the following exact queuing theory:

- For a single-server system, the model is compared to that of the exact queuing theory by Bolch *et al.* [26]. In this case, the relationship between the upper delay limit and the error of violation is:

$$W = \frac{\log(\varepsilon)}{-\mu(1-\rho)} \tag{8}$$

where W is the delay, ε is the error rate, μ is the service rate, and ρ is the server utilization rate.

- For multi-server systems, the model is compared to that of the exact theory proposed by F. Ciucu *et al.* [27]. The relationship between the upper delay limit and the error of violation rate is given as follows:

$$P(W > w) = \sum_{i=0}^{n-1} \frac{\mu\left((1-\rho)w\right)^i}{i!} e^{-\mu(1-\rho)w} \tag{9}$$

where W is the delay, w is the delay threshold, μ is the service rate, ρ is the server utilization rate, n is the number of servers, and i is the i-th server.

When Equation (9) is used under our simulation conditions with a single OLT as the only server ($n = 1$), it becomes:

$$P(W > w) = \mu e^{-\mu(1-\rho)w} \tag{10}$$

And W follows an exponential distribution with parameter $\varepsilon(\mu(1-\rho))$. Thus, the Equations (8) and (10) from the exact queuing theory will be subjected to the calculations performed in section 3 with our dataset. The various curves obtained will be compared to that of our model in order to first observe the conformity of its shape and subsequently determine the value of the upper delay limit as a function of certain queuing parameters.

In the literature, two violation error parameters are adopted to reflect the requirements of the service-level agreement (SLA). These are $\varepsilon = 10^{-6}$ or $\varepsilon = 10^{-4}$; however, we vary this violation error rate from 10^{-7} to 1, as in [14], to observe the behavior of the delay. On the graphs, the delay curves are plotted as a function of parameters such as the violation error rate, the service rate, and the server utilization rate, as presented below.

5.1. The Impact of the Service Rate on the Upper Delay Limits

We know that bandwidth allocation impacts the delay in the XG-PON network; thus, we aim to observe the behavior of bandwidth on the delay beyond the thresh-

old (1.5 ms) prescribed by the ITU.

Upon observing the curves in Figure 3 and Figure 4, the curve of our model (blue) has a similar shape to those of the exact theory (red). Both curves in each figure exhibit a significant decrease in the delay limit, from 0.003 seconds to 0.0015 seconds for our model and from 0.003 seconds to 0.0005 seconds for the exact theory models. Both curves appear to stabilize when they reach the delay limit of 0.0015 seconds for one and 0.0005 seconds for the other.

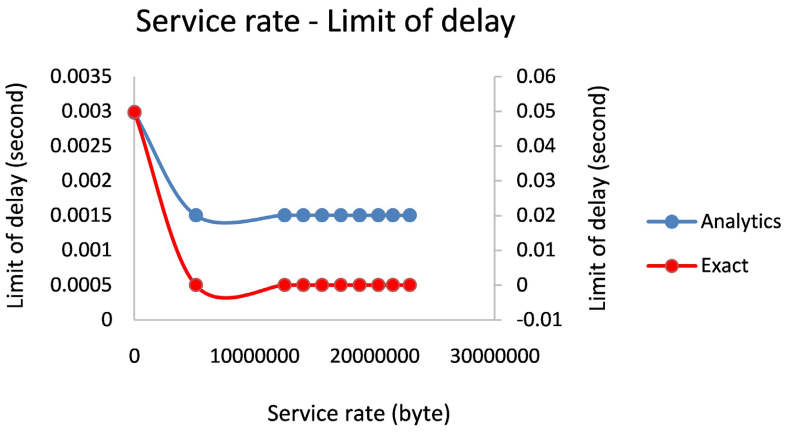

Figure 3. Service rate on the upper limit of the delay: Case for 1 server.

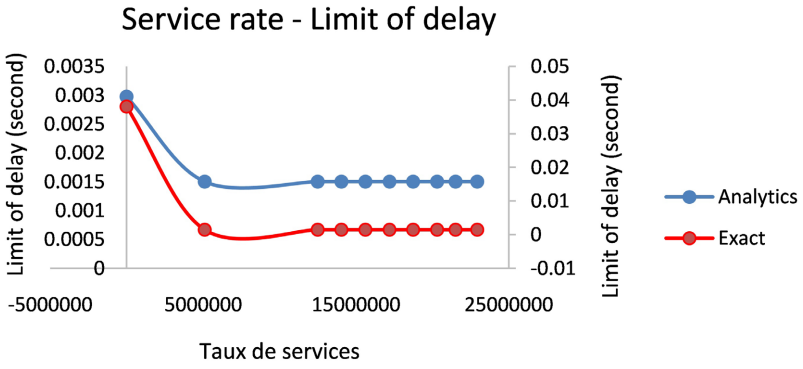

Figure 4. Service rate on the upper limit of the delay: Case for n servers.

5.2. The Server Utilization Rate and the Upper Limit of the Delay

In Figure 5 and Figure 6, we investigate the impact of server load on the upper delay limit. To do this, we represent the curve of our model (blue) and those of the exact theory (red) by varying the server utilization rate.

The observation of the server utilization rate and the upper delay limit is made on two figures:

In Figure 5 (without gridlines), we observe the curve of our model (blue) and that of the exact theory (red) for a single server, as proposed by Bolch *et al.*, with model Equation (8).

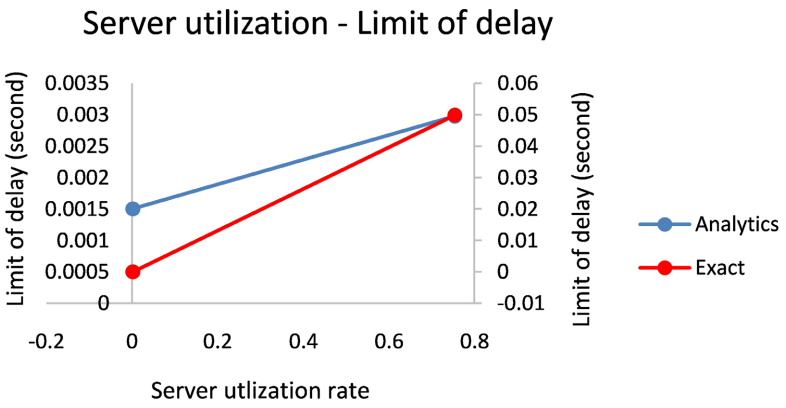

Figure 5. Rate of use of the server and the upper limit of the delay: Case for 1 server.

In **Figure 6** (with gridlines), it is still the curve of our model (blue) and that of the exact theory (red) for n servers, with the model proposed by F. Ciucu *et al.* in Equation (10).

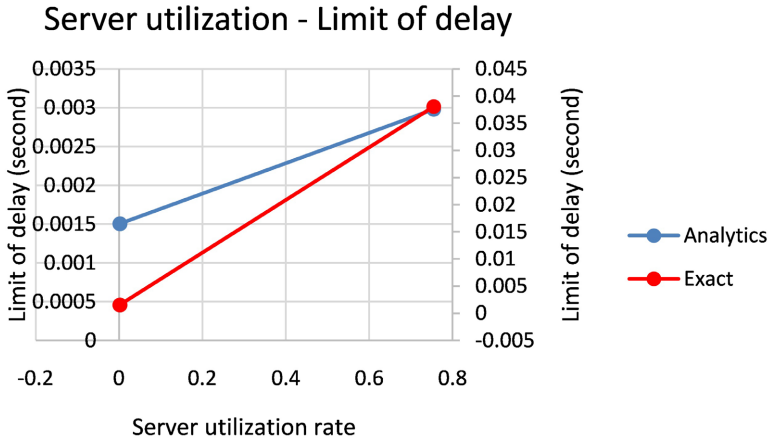

Figure 6. Rate of use of the server and the upper limit of the delay: Case for n servers.

Whether in **Figure 5** or **Figure 6**, the curve of our model (blue) has the same shape as those of the exact theory (red).

In both figures, our curve starts with an increase in the delay threshold, reaching 0.0015 seconds before reaching a delay of 0.003 seconds.

In contrast, the exact theory curves increase from 0.0005 seconds to also reach the value of 0.003 seconds.

5.3. The Violation Error and the Upper Limit of the Delay

In these figures, we study the performance of the models to meet the requirements of the service-level agreement (SLA) in terms of violation error and delay limitation.

Figure 7 and **Figure 8** show the upper delay limit while varying the violation

error. In Figure 7 (without gridlines), we observe the behavior of the exact theory curve for a single server alongside our model, and similarly for the exact theory curve with n servers and ours in Figure 8 (with gridlines).

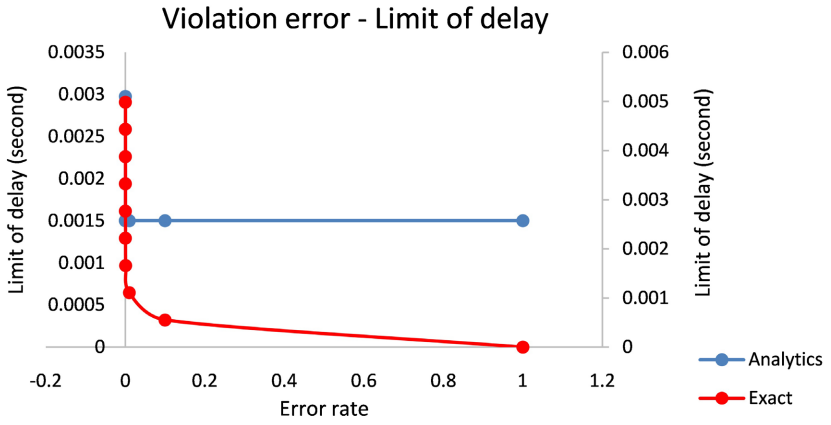

Figure 7. The violation error on the upper limit of the delay: Case for 1 server.

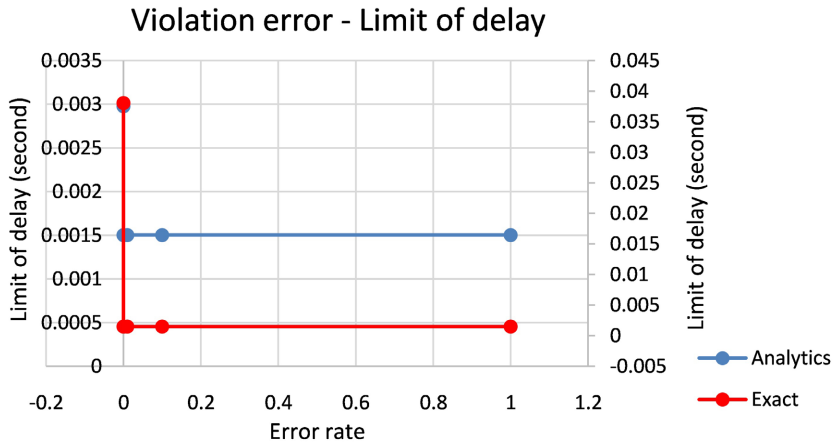

Figure 8. The violation error on the upper limit of the delay: Case for n servers.

In Figure 7 and Figure 8, the curves decrease as the violation error rate increases. The decrease starts from the point with coordinates (0.0000001; 0.003), which corresponds to a low error value and the upper delay limit, down to the point with coordinates (0.00001; 0.0015) for our model and to the point with coordinates (0.01; 0.0005) for the curves of the exact theory. Beyond these points, the curves either stabilize up to 1 (Figure 8) or reduce the decrease down to 1 (Figure 7).

6. Performance Analysis

The delay curves are plotted as a function of:

Service rate (μ)

Server utilization $\rho = \lambda/\mu$

Violation error P (delay > 3 ms) or threshold exceedance probabilities.

In Figure 3 and Figure 4 (service rate (μ) and the upper bound of delay), the delay of the analytical model generally decreases like $1/\mu$, and the effect of ρ is reflected via λ/μ; as μ increases, delays decrease and the bound is dominated by the mean service time (0.0015 second).

Observation: Rapid decrease of delay up to a certain value of μ, then attenuation of the effect when λ approaches μ (increase of ρ).

For models from exact theory the delay follows a similar decay at the head of the curve, but the inflection and slope can be modulated by overheads, grant cycles and medium contention.

For higher μ, the gap between the curves can shrink, since the relative impact of contention delays and overheads becomes less important when service is fast compared to arrivals.

In other words, the service rate represents bandwidth allocation. We note that when bandwidth is not provided (zero service rate), delay reaches its upper bound for both curves. When the Optical Line Terminal (OLT) triggers bandwidth allocation, the upper bound of our model decreases until it stabilizes at the mean delay (1.5 ms), a threshold prescribed by the International Telecommunication Union (ITU). This result confirms the performance of our model since we are seeking the behavior of the delay beyond the threshold. For the exact-theory model, the delay limit not to be exceeded is always 0.003 second; however, the curve decreases below the 0.0015 second threshold, which is an acceptable zone according to our initial assumptions.

The gaps between the curves reflect the fact that at low μ, the analytical model can underestimate the network's sensitivity, whereas the exact models may show even more sensitive delays.

At high μ, the curves often converge; overheads become insignificant compared with fast service, which reduces the gaps.

In Figure 5 and Figure 6, we observe the behavior of the upper bound as a function of server utilization.

For the analytical model the delay curve increases monotonically with ρ and diverges when ρ approaches 1.

Sensitivity increases rapidly near saturation and depends strongly on μ and λ.

In the 1.5 ms to 3 ms interval: for given μ, there exists a range of ρ where the mean delay ≈ 1.5 ms; beyond that, it climbs and can exceed 3 ms if ρ is sufficiently high.

For exact theory, the mean delay is generally higher than the analytical model at the same ρ, especially when medium contention and overheads exist (grant cycles, synchronization, arbitration).

Queues can become longer and more variable; the curve can show a more pronounced curvature, with delays rising faster under heavy load.

In light-load regime (ρ low to ~0.5), the mean-delay curves from both frameworks are close; the gap is small (a few tenths of a millisecond to a few hundred μs).

In moderate to high regime ($0.5 < \rho < 0.9$), the analytical model underestimates the mean delay compared to exact theory.

In heavy regime ($\rho > 0.9$), the mean delay from exact theories significantly exceeds the analytical estimate, due to long queues and access periods that introduce additional delays.

The shape of the curves shows that regardless of the model, the more the OLT is stressed, the more delay increases because bandwidth becomes scarce as the server is solicited to allocate it.

Figure 7 and Figure 8 show the upper bound of delay while varying the error rate.

For the analytical model, the error rate or the probability that delay exceeds 3 ms $P(D > 3 \text{ ms})$ increases with ρ; for low ρ, the probability is small and under heavy load it grows rapidly.

For the exact-theory models, $P(D > 3 \text{ ms})$ is often higher than in the analytical framework, particularly under heavy load and with multi-flow contention.

This is because TDMA mechanisms and access cycles introduce longer queues and latency spikes.

We observe that the smaller the rate, the larger the upper bound of delay. Indeed, the violation error rate that reflects the requirements of the service-level agreement is stricter when it is low. In other words, the more restrictions (low rate), the longer packet waiting times. This explains the upper bound of delay that is reached at 0.003 seconds when the violation error rate is 10^{-7}.

Furthermore, when there are fewer restrictions (high rate), packet waiting time is short and delay decreases (Figure 7 and Figure 8). The threshold rate for our model is 10^{-5}, since at this value we reach the delay threshold of 0.0015 second.

At the end of our analysis, we conclude that for operational use the models should be used complementarily:

- The analytical model for a quick estimate, bounds and intuition on trends as a function of λ and μ.

- Exact-theory models to assess SLA safety margins and to study realistic scenarios with TDMA, overheads and contention.

7. Conclusion

This study aims to establish a reliable delay bound for real-time services over XG-PON networks, to ensure quality of experience for critical applications (videoconferencing, telemedicine, IPTV) and to guide network design and operation. To this end, we mobilized an approach integrating analytical modeling and statistical estimations linking the maximum acceptable delay to QoS metrics and traffic distributions, complemented by queuing models and simulations that validate the bound under various loads, topologies and bandwidth allocation policies. Furthermore, we explore the potential contribution of deep learning techniques to estimate in real-time the probability of violation and to guide proactive adjustments of network parameters, in particular in the context of DBA and dynamic

SLAs. The results show that the 3 ms bound can serve as an operational reference under controlled conditions, with margins and limits depending on traffic variations and flow priorities; Sensitivity analyses suggest the usefulness of adaptive mechanisms and predictive monitoring to maintain compliance. Finally, the operational impact translates into continuous monitoring of SLAs, adaptive optimization of DBA and SLA parameters, as well as ONU and link sizing guidelines, which enable dynamic allocation of resources while preserving critical deadlines and improving the user experience. Beyond determining this limit, we identified the parameters that can be adjusted, along with their values, to regulate the delay: setting the violation error rate to 10^{-7} allows reaching the upper delay limit, while a violation error rate of 10^{-5} enables reaching the delay threshold.

Future research could expand on this work by Study how theoretical delay bounds evolve when network architecture or bandwidth allocation mechanisms change (e.g. introduction of advanced techniques like Flow Prolongation or ML approaches for de-lay prediction); use machine learning to predict the bounds of the delay, aiming for greater accuracy; test compliance with the 3 ms delay limit for real-time services (video conferencing, telemedicine, IPTV) in XG-PON; integrate neural networks to model the delay using a set of input parameters in the ONUs.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Toral-Cruz, H., Pathan, A.K. and Ramírez Pacheco, J.C. (2013) Accurate Modeling of VoIP Traffic QoS Parameters in Current and Future Networks with Multifractal and Markov Models. *Mathematical and Computer Modelling*, **57**, 2832-2845. https://doi.org/10.1016/j.mcm.2011.12.007

[2] Memon, K.A., Mohammadani, K.H., Ain, N.U., Shaikh, A., Ullah, S., Zhang, Q., *et al.* (2019) Demand Forecasting DBA Algorithm for Reducing Packet Delay with Efficient Bandwidth Allocation in XG-PON. *Electronics*, **8**, Article 147. https://doi.org/10.3390/electronics8020147

[3] Mikaeil, A.., Hu, W.S., Hussain, S.B., *et al.* (2018) Traffic-Estimation-Based Low-Latency XGS-PON Mobile Front-Haul For Small-Cell C-RAN Based on an Adaptive Learning Neural Network. *Applied Sciences*, **8**, 1097.

[4] Akanza Konan Ricky, N., Bi Tra, G., Jean Edgard, G. and Yve, T. (2022) A Dynamic Bandwidth Allocation System in XG-PONs Based on Statistical Estimation of Buffer Inputs and Outputs. *International Journal of Advanced Research*, **10**, 544-552. https://doi.org/10.21474/ijar01/15522

[5] Gore, R.N., Lisova, E., Åkerberg, J. and Björkman, M. (2022) Network Calculus Approach for Packet Delay Variation Analysis of Multi-Hop Wired Networks. *Applied Sciences*, **12**, Article 11207. https://doi.org/10.3390/app122111207

[6] Hood, D. (2012) Gigabit-Capable Passive Optical Networks. John Wiley & Sons Inc., Hoboken, New Jersey.

[7] Geleji, G. and Perros, H. (2014) Jitter Analysis of an MMPP-2 Tagged Stream in the Presence of an MMPP-2 Background Stream. *Applied Mathematical Modelling*, **38**,

3380-3400. https://doi.org/10.1016/j.apm.2013.11.055

[8] Dbira, H., Girard, A. and Sansò, B. (2016) Calculation of Packet Jitter for Non-Poisson Traffic. *Annals of Telecommunications*, **71**, 223-237. https://doi.org/10.1007/s12243-016-0492-0

[9] Thombre, S. (2018) Network Jitter Analysis with Varying TCP for Internet Communications. 2018 3*rd International Conference for Convergence in Technology* (*I2CT*), Pune, 6-8 April 2018, 1-7. https://doi.org/10.1109/i2ct.2018.8529816

[10] Fei, N., Xuefen, C., Wen, D. and Haifeng, Y. (2016) Jitter Analysis of Real-Time Services in IEEE 802.15.4 WSNs and Wired IP Concatenated Networks. *The Journal of China Universities of Posts and Telecommunications*, **23**, 1-8. https://doi.org/10.1016/s1005-8885(16)60039-0

[11] Lubben, R., Fidler, M. and Liebeherr, J. (2014) Stochastic Bandwidth Estimation in Networks with Random Service. *IEEE/ACM Transactions on Networking*, **22**, 484-497. https://doi.org/10.1109/tnet.2013.2261914

[12] Yang, G., Xiao, M., Al-Zubaidy, H., Huang, Y. and Gross, J. (2018) Analysis of Millimeter-Wave Multi-Hop Networks with Full-Duplex Buffered Relays. *IEEE/ACM Transactions on Networking*, **26**, 576-590. https://doi.org/10.1109/tnet.2017.2786341

[13] Azuaje, O. and Aguiar, A. (2019) End-to-End Delay Analysis of a Wireless Sensor Network Using Stochastic Network Calculus. 2019 *Wireless Days* (*WD*), Manchester, 24-26 April 2019, 1-8. https://doi.org/10.1109/wd.2019.8734241

[14] Miao, W., Min, G., Wu, Y., Huang, H., Zhao, Z., Wang, H., *et al.* (2019) Stochastic Performance Analysis of Network Function Virtualization in Future Internet. *IEEE Journal on Selected Areas in Communications*, **37**, 613-626. https://doi.org/10.1109/jsac.2019.2894304

[15] Wang, H., Wu, Y., Min, G. and Miao, W. (2022) A Graph Neural Network-Based Digital Twin for Network Slicing Management. *IEEE Transactions on Industrial Informatics*, **18**, 1367-1376. https://doi.org/10.1109/tii.2020.3047843

[16] Bondorf, S. (2017) Better Bounds by Worse Assumptions—Improving Network Calculus Accuracy by Adding Pessimism to the Network Model. 2017 *IEEE International Conference on Communications* (*ICC*), Paris, 21-25 May 2017, 1-7. https://doi.org/10.1109/icc.2017.7996996

[17] Geyer, F., Scheffler, A. and Bondorf, S. (2022) Network Calculus with Flow Prolongation—A Feedforward FIFO Analysis Enabled by ML. arXiv: 2202.03004. http://arxiv.org/abs/2202.03004

[18] Sahu, M., Damle, S. and Kherani, A.A. (2021) End-to-End Uplink Delay Jitter in LTE Systems. *Wireless Networks*, **27**, 1783-1800. https://doi.org/10.1007/s11276-020-02517-7

[19] Mansour, Y. and Patt-Shamir, B. (2002) Jitter Control in QoS Networks. *IEEE/ACM Transactions on Networking*, **9**, 492 502. https://doi.org/10.1109/90.944346

[20] Dbira, H. (2017) Analyse mathématique, méthode de calcul de la gigue et applications aux réseaux Internet. Master's Thesis, École Polytechnique de Montréal.

[21] Nichols, K., Jacobson, V. and Poduri, K. (1999) An Expedited Forwarding PHB. Internet Engineering Task Force, Request for Comments RFC 2598.

[22] Poretsky, S., Erramilli, S., Perser, J. and Khurana, S. (2006) Terminology for Benchmarking Network-Layer Traffic Control Mechanisms. Internet Engineering Task Force, Request for Comments RFC 4689.

[23] Little, J.D. (2011) OR FORUM—Little's Law As Viewed on Its 50th Anniversary. *Operations Research*, **59**, 536-549. https://doi.org/10.1287/opre.1110.0940

[24] Angelopoulos, J.D., Leligou, H., Argyriou, T., Zontos, S., Ringoot, E. and Van Caenegem, T. (2004) Efficient Transport of Packets with QOs in an FSAN-Aligned GPON. *IEEE Communications Magazine*, **42**, 92-98. https://doi.org/10.1109/mcom.2003.1267106

[25] (2024) Modélisation d'une file d'attente. https://math.univ-lyon1.fr/~alachal/serveurOT/files_attente.pdf

[26] Bolch, G., Greiner, S., de Meer, H. and Trivedi, K.S. (2006) Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications. Wiley. https://doi.org/10.1002/0471791571

[27] Ciucu, F. (2007) Network Calculus Delay Bounds in Queueing Networks with Exact Solutions. In: Mason, L., Drwiega, T. and Yan, J., Eds., *Managing Traffic Performance in Converged Networks*, Springer, 495-506. https://doi.org/10.1007/978-3-540-72990-7_45