Revista Brasileira de Computação Aplicada, July, 2025
DOI: 10.5335/rbca.v17i2.16481
Vol. 17, Nº 2, pp. 64–77
Homepage: seer.upf.br/index.php/rbca/index

ARTIGO ORIGINAL

Performance Analysis of Ethernet Networks Through Quality of Service (QoS) Metrics Using Real and Virtual Machines

Diones da Silva Rodrigues [1] and Diógenes Antonio Marques José [1]

[1]Mato Grosso State University (UNEMAT), Brazil

*diones.silva@unemat.br; dioxfile@unemat.br

Received: 2024-11-13. **Revised:** 2025-07-25. **Accepted:** 2025-07-31.

Abstract

Quality of Service (QoS) encompasses technologies used in computer networks to ensure reliable application performance. Consequently, the main QoS performance metrics are throughput, end-to-end delay, *jitter*, and packet loss rate. One problem regarding computer networks is finding an ideal environment that guarantees the minimum QoS requirements, especially in a *home office*. Therefore, this work analyzed the performance of a computer network in two contexts, with and without virtual machines. Thus, *Iperf3* was used to generate traffic in two ways: originating from an *Iperf3* client on a real machine, and originating from an *Iperf3* client installed on a virtualized *OS*. The tests used *UDP* connections to two Iperf3 servers: local (LAN) and remote (WAN). The goal of this was to differentiate services that require high throughput from those that require minimal delay. The QoS metrics were throughput, end-to-end delay, jitter, and packet loss rate. The results highlighted the importance of minimizing packet loss to ensure efficient communication, especially in services such as e-mail delivery and file transfer.

Keywords: Iperf3; Network Performance Metrics; Quality of Service (QoS).

Resumo

Qualidade de Serviço (QoS) engloba tecnologias utilizadas em redes de computadores para garantir um desempenho confiável de aplicativos. Consequentemente, as principais métricas de desempenho de QoS são a vazão, o atraso, o *jitter* e a taxa de perda de pacotes. Um problema com relação às redes de computadores é encontrar um ambiente ideal que garanta os requisitos mínimos de QoS, principalmente no *home office*. Portanto, este trabalho analisou o desempenho de uma rede de computadores em dois contextos, com e sem máquinas virtuais. Dessa forma, foi utilizado o *Iperf3* para gerar tráfego de duas maneiras: originado de um cliente *Iperf3* em uma máquina real; e originado de um cliente *Iperf3* instalado em um *SO* virtualizado. Os testes utilizaram conexões *UDP* para dois servidores Iperf3: local (LAN) e remoto (WAN). O objetivo disso foi diferenciar serviços que requerem alta vazão dos que necessitam de atraso mínimo. As métricas de QoS utilizadas foram vazão, atraso, *jitter* e taxa de perda de pacotes. Os resultados destacaram a importância de minimizar a perda de pacotes para garantir uma comunicação eficiente, especialmente em serviços como entrega de *e-mails* e transferência de arquivos.

Palavras-Chave: Iperf3; Métricas de Desempenho de Redes; Qualidade de Serviço (QoS).

1 Introduction

With the high demand for Internet access and the expansion of systems and services that rely on the network, QoS has become an essential factor for user satisfaction. Furthermore, with the increase in telecommuting during the pandemic, the network has become even more critical, as many activities that were previously performed in person now depend on the Internet, such as virtual meetings and file transfers. QoS refers to the network's ability to deliver a service that meets users' expectations regarding availability, performance, reliability, and security.

According to the International Organization for Standardization (ISO), QoS is defined as a set of traffic performance factors that determine the user's satisfaction level with the service the telecommunications operator offers (ISO/IEC13236, 1998). In this context, evaluating service quality is vital due to the increase in remote work and home offices due to the COVID-19 pandemic. Therefore, this study highlights the importance of considering the increase in bandwidth during pandemic periods. According to the Organization for Economic Cooperation and Development (OECD) (2020), there was a significant increase of 22.3% in the bandwidth used by the global population between December 2019 and March 2020, more than four times that of the previous quarter. In Germany, for example, there was an increase from 11.2% to 16.5%, while in Italy, there was a consumption of 39.9% more bandwidth during this period, compared to a growth of only 1.8% in the previous quarter. According to the OECD, regions such as Japan, Chile, the United States, Singapore, South Africa, and Brazil also reported similar trends (OECD, 2020).

Another interesting issue, according to the (Lee et al., 2021), during the pandemic, there was an increased demand from Americans for the quality of network services. With the need for remote work, 81% of Americans engaged in video conferencing, and 40% stated that digital tools have become essential in their professional lives. As the pandemic progressed, they realized the need to upgrade their Internet access services to meet the increased traffic demand. Consequently, 29% of broadband users took steps to improve the speed, reliability, or quality of their Internet connection (Lee et al., 2021). Therefore, this study aimed to perform a performance analysis of a controlled network with two traffic sources, using real and virtual machines, using a public server hosted in Brazil, specifically in São Paulo (speedtest.sao1.edgoo.net). This shared server is available free of charge for the community to perform tests, and other servers can be found on the GitHub page (Community, 2024). The traffic will be generated by the Iperf3 tool using the UDP protocol.

Additionally, two environments were used for comparison: Environment 1, consisting of real machines, and Environment 2, consisting of virtual machines. The objective was to identify the differences in QoS between the two approaches. Furthermore, these results will also be used to generate a usage profile for each service, for example, differentiating services requiring higher throughput from those requiring minimal delay (ToS). The QoS metrics used in this investigation were throughput, end-to-end delay (latency), jitter, and packet loss rate.

The results showed significant differences between the two studied environments. Both environments presented low jitter values, indicating stable audio and video transmission on demand. However, Environment 2 demonstrated notably superior performance in terms of packet loss rate, approaching zero. Regarding latency, no significant differences were found between the two environments. These results are considered adequate for most remote access applications, video calls, and web browsing.

The remainder of this paper is organized as follows: Section 2 presents Related Work; Section 3 shows the Proposal Description, the Materials and Methods for developing the study, and the experiments conducted; Section 4 discusses the QoS Metrics; Section 5 presents the Iperf3 tool; Section 6 presents the Results and Discussion; and finally, Section 7 offers the conclusions and future works.

2 Related Work

Facchini et al. (2020) presents the results of an experimental study on multi-cast video traffic in data networks, which aimed to evaluate performance and QoS under limited bandwidth conditions. The experimental results, analyzed based on relevant metrics, showed that despite increasing video traffic consumption, it is possible to efficiently utilize available resources without compromising QoS.

As described in Mazhar et al. (2023), the performance analysis of a wireless sensor network using QoS metrics is presented. The study describes traffic modeling using traffic engineering to improve resource allocation in the network. The study results show that implementing QoS can significantly improve network latency and packet loss rate, even in situations with heterogeneous traffic. The article highlights the importance of implementing QoS in wireless sensor networks and provides a practical approach to optimizing resource allocation in the network.

As presented in Valencia et al. (2020), the need to ensure the quality of service (QoS) in networks supporting video streaming services is discussed, and how the software-defined networking (SDN) architecture can help achieve this goal. The study evaluated QoS metrics of a real and emulated SDN network for a television streaming service (IPTV) using the RTMP and RTSP transmission protocols. The results showed that RTMP offers more excellent stability in delivering multimedia content and that SDN can match the performance of conventional architecture. Still, the protocol choice can affect QoS metrics such as end-to-end delay, jitter, packet loss rate, and noise level.

Another article related to traffic management is Reisslein et al. (1999), which discusses the management of multimedia traffic that can tolerate some loss but has strict delay constraints. A natural requirement of QoS for a multimedia connection is a prescribed limit on the fraction of traffic that exceeds an end-to-end delay limit. The article proposes and analyzes a traffic management scheme that guarantees QoS for multimedia traffic while simultaneously allowing a large connection capacity. The paper shows that the loss probability is minimized with simple one-buffer smoothers operating at specific minimal rates. The proposed scheme is based on worst-case traffic and can be used to guarantee QoS for regulated traffic.

The work described in da Costa (2008) addresses the implementation of QoS in IP networks through performance metrics. The work highlights the importance of using pre-established metrics in RFCs 2544, 2889, and 3918, respecting the validation and result disclosure methods. Based on the analysis performed by the author, primary and complementary references were determined for performance evaluation in QoS networks, and the importance of using a dedicated operating system to avoid compromising test results due to shared processing by other processes was also emphasized.

In Wang et al. (2014), a model for evaluating Quality of Experience (QoE) based on QoS for video services in communication networks is presented. The experimental results conducted by the author show that QoE can be obtained from QoS parameters in network communications, which is vital for optimizing bandwidth allocation and predicting QoE for high-definition video based on QoS parameters.

In the study described in Dhafer R. Zaghar (2013), the importance of QoS as a measure of efficiency in Ad-hoc networks was analyzed, considering their complexity and variation over time. The experimental result showed that higher throughput only sometimes means high QoS, and the proposed new system exhibited superior behavior to the traditional method and can also be used to evaluate the performance of MANET protocols based on user needs and provide a unique QoS value to end-users.

In Khamosh et al. (2023), the relationship between quality of service (QoS) parameters - packet loss, latency, and jitter - and the quality of experience (QoE) of Internet of Things (IoT) services were investigated. A subjective evaluation approach established a connection between subjective opinion scores and QoS variables. Additionally, a mapping model from QoS to QoE was proposed. This research indicated a close relationship between these variables, opening possibilities for further research on the quality of experience of IoT services. The main objective of this study was to investigate the effect of QoS parameters - packet loss, latency, and jitter - on the quality of experience of IoT services and to propose a mathematical model to predict QoE based on the quality of service factors.

In Kesavan et al. (2023), a proportionally fair resource allocation algorithm for device-to-device (D2D) communication in fifth-generation (5G) wireless communication networks is proposed. The goal was to design an algorithm that maintains QoS for cellular users (CUs) while providing D2D communication. The results showed that the algorithm provides fairness and high throughput for D2D users without significantly impairing the throughput of CUs.

The study described in Nazia Tabassum (2022) explored the distribution of data and security aspects of the Internet of Things Vehicular with Cloud Computing (IoV-CC), considering different computing approaches. The results obtained in this study highlighted the importance of IoV in providing efficient technology for autonomous driving, vehicle control, and intelligent systems. Furthermore, the integration between VANET and Cloud Computing significantly improves QoS, reduces congestion, increases road safety, and offers alternative services to drivers.

3 Proposal Description

This study aims to analyze the performance of a controlled network with two traffic sources generated from real and virtual machines using a public server hosted in Brazil, such as speedtest.sao1.edgoo.net (e.g., two environments were used: Environments 1 (Fig. 1) and 2 (Fig. 2)). The main objective is to identify differences in QoS between traffic generated from virtual and real machines. These results will also be used to create a usage profile for each service, distinguishing between services that require higher throughput and those that require minimal delay (ToS).

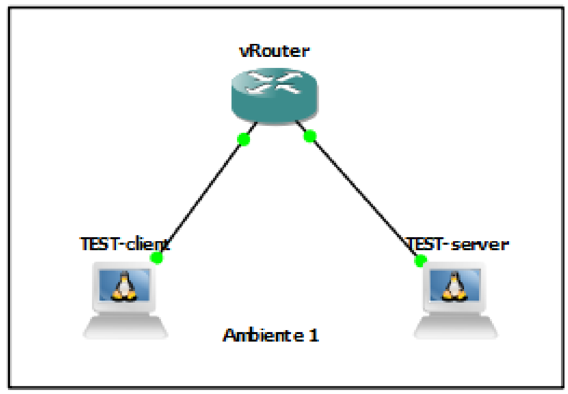

Figure 1: Environment 1, where the real machines, client, and local server were tested.

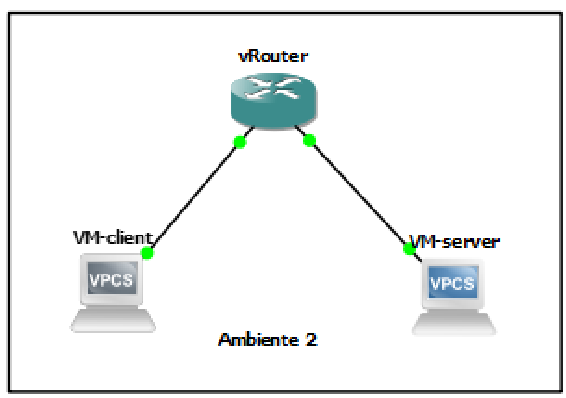

Figure 2: Environment 2, where the virtual machines, client, and local server were tested.

3.1 Materials and Methods

In this study, a case study was conducted to gather network evaluation requirements for virtual and physical machines. Therefore, two environments were developed to assess

QoS metrics such as end-to-end delay, jitter, throughput, and packet loss rate. Test Environments 1 and 2 were set up using real machines with the following configuration: 8 GB of RAM, eight cores, and 16 threads, all connected with Gigabit Ethernet cards. Each machine ran the 64-bit Linux Debian 11 operating system. In addition, these machines were configured to minimize interference from other services, ensuring optimal test conditions. The objective of this configuration was to identify performance differences between real and virtual machines regarding QoS metrics: throughput, end-to-end delay, jitter, and packet loss rate.

In the testing phase, experimental research was conducted to analyze the QoS metrics using the Iperf3 tool. The results were used to generate graphs and facilitate data analysis. The experimental procedure involved the development of two environments to evaluate QoS metrics and two test types (e.g., Local Server and Public Server). Environment 1 consisted of real machines, while Environment 2 comprised virtual machines. The virtual machines were created and executed using the *VMware Workstation Player 17* virtualization tool in the non-commercial version. Both environments were tested, connected to a local area network (LAN) (e.g., 192.168.1.0/24) and a remote network via *speedtest.sao1.edgoo.net*. Consequently, each test was conducted at different periods to avoid interference between the tests.

In Environment 1 (Local Server), two real machines were used, as shown in Fig. 1. The client has the IP address 192.168.1.101, while the server has the IP address 192.168.1.100. This was the command used for testing in the environment, from the client machine to the server:

```
iperf3 -c 192.168.1.100 -t 30 -i 1 -u
-b 800M
```

- "-c 192.168.1.100": specifies that the client (the one executing the command) will connect to the server with the IP address 192.168.1.100. This is the machine with which the performance test will be conducted.
- "-t 30": sets the test duration in seconds. In this case, the test was run for 30 seconds. It indicates the time of the test.
- "-i 1": sets the interval for displaying results. In this case, the results will be displayed every 1 second. This means that every second, Iperf3 will display metrics during the test.
- "-u": specifies that the test will be conducted using the UDP protocol.
- "-b 800M": sets the maximum bandwidth (transfer rate) for the test. In this case, the value is 800 Mbps (megabits per second). This parameter limits the bandwidth used in the test, simulating a specific scenario where the connection is limited to this rate.

In Environment 2 (Local Server), virtual machines were used, as shown in Fig. 2. The client has the IPv4 address 192.168.1.201, and the server has the IPv4 address 192.168.1.200. A bandwidth of 800Mbps was used. This was the command used for testing in the environment, from the client machine to the server:

```
iperf3 -c 192.168.1.200 -t 30 -i 1 -u
-b 800M
```

- Idem.

The test conducted in Environment 1 (Public Server) was performed using a physical machine tested on a public server. The client with the IPv4 address 192.168.1.101 was located on a local network, while the public server was on a remote network, speedtest.sao1.edgoo.net. The test was conducted with a bandwidth of 800Mbps; the traffic modeling can be seen in Fig. 3.

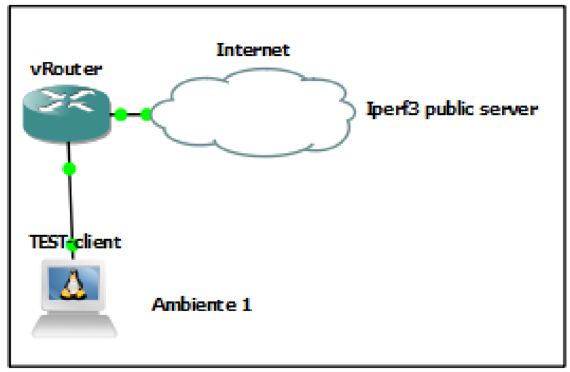

Figure 3: Environment 1, where the public server was tested with a physical machine.

The command used for testing in the client machine's environment is sent to the public server:

```
iperf3 -c speedtest.sao1.edgoo.net -p
9201-9240 -t 30 -i 1 -u -b 800M
```

- "-c speedtest.sao1.edgoo.net": specifies that the client (who is running the command) will connect to the server with the address
- "speedtest.sao1.edgoo.net". This is the server with which the QoS test will be performed.
- "-p 9201-9240": sets the range of ports for communication between the client and the server. In this case, ports 9201 to 9240 will be used.
- "-t 30": defines the test duration in seconds. In this case, the test will run for 30 seconds. This indicates the time of the test.
- "-i 1": sets the interval for displaying the results. In this case, the results will be displayed every 1 second.
- "-u": specifies that the test will be performed using the UDP protocol.
- "-b 800M": defines the maximum bandwidth (transfer rate) for the test. In this case, the value is 800 Mbps (megabits per second). This parameter limits the bandwidth used in the test, simulating a specific scenario where the connection is limited to this rate.

The test conducted in Environment 2 (Public Server) utilized a virtual machine tested on a public server, as shown in Fig. 4. The client has the IPv4 address 192.168.1.101, and the public server has the address: speedtest.sao1.edgoo.net. This public server is located in São Paulo, Brazil (Community, 2024). A bandwidth of 800Mbps was used, just like in the other tests.

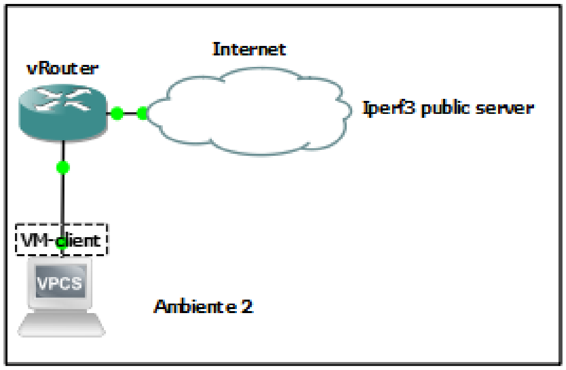

Figure 4: Environment 2, where the public server was tested using a virtual machine.

The command used for testing in Environment 2, from the client machine to the public server, was:

```
iperf3 -c speedtest.sao1.edgoo.net -p
9201-9240 -t 30 -i 1 -u -b 800M
```

- Idem.

4 QoS Metrics

4.1 Throughput

The throughput is the average rate at which the network successfully sends and receives data. It is the number of bits sent from a source node to a destination node divided by the observation time duration on a specific network link (Forouzan, 2007; Susom, 2018).

$$Throughput = Bits/time(s). \tag{1}$$

4.2 End-to-End Delay (Latency)

Latency, or end-to-end delay, is defined as the time it takes for a packet to travel from the source to the destination and is composed of all the network delays from the moment the packet leaves the application layer of the source node until it arrives at the application layer of the destination node. In other words, latency is the time between sending and receiving a data packet (Duc et al., 2016; Forouzan, 2007).

$$Delay = (Packet_{Time\ destiny} - Packet_{Time\ origin})(sec). \tag{2}$$

4.3 Jitter

Jitter is the variation in packet delay within a network, also called fluctuation (Tanenbaum and Wetherall, 2011).

$$Jitter = |\mathrm{Delay}_n - \mathrm{Delay}_{n-1}|(sec). \tag{3}$$

4.4 Packet Loss Rate

The packet loss rate is one of the most critical metrics impacting service quality and performance. The packet loss rate measures network reliability itself (Forouzan, 2007; Villarim et al., 2023).

$$PLR = \text{All Packets Sent} - \text{All Packets Received}. \tag{4}$$

Additionally, the packet loss rate (PLR) can be derived from the packet delivery rate formula described in (Villarim et al., 2023). Specifically, PLR is given by the Eq. (5):

$$PLR = \left(1 - \frac{Z_{RX}}{Z_{TX}}\right) * 100. \tag{5}$$

Where Z_{TX} represents the total number of transmitted packets and Z_{RX} is the total number of received packets.

5 Iperf3

The Iperf3 traffic generator is an open-source tool developed by NLANDR/DAST (National Laboratory for Applied Network Research/Distributed Applications Support Team) and utilized by network professionals to assess the network quality of a connection through TCP and UDP protocols (Dugan et al., 2024).

With Iperf3, it is possible to measure a network connection's bandwidth, latency, jitter, and packet loss. Iperf3 is compatible with various operating systems, including Windows, Linux, and MacOS. The use of Iperf3 as a network evaluation tool for QoS has been extensively explored in several studies, for example, (Henrique and Alves, 2014; Agusriandi and Elihami, 2020; Pratama and Wikantyasa, 2019).

Iperf3 is a command-line and parameter-driven application, for example:

- c - IP address of the Iperf3 server to be tested;
- p - Port number of the Iperf3 server to be tested;
- t - Test duration in seconds;
- i - Interval between performance measurements;
- u - Use UDP protocol instead of TCP;
- b - Bandwidth in bits per second.

6 Results and Discussion

This section presents the results and discussion of the experiments conducted to evaluate the QoS metrics in two distinct environments: Environment 1, composed of physical machines, and Environment 2, consisting of virtual machines.

The objective of these tests was to analyze and compare the performance of the metrics in each environment, providing a comprehensive understanding of the impact of QoS metrics in different configurations.

This section will present the results for each tested metric in both environments, including additional tests conducted on the public server. Furthermore, the results will be directly compared, highlighting the observed differences.

6.1 Results of Environment 1 tests on the local server

In this subsection, the graphs of the results for each metric in Environment 1 with a local server will be presented.

The individual results for each metric can be seen in Fig. 5, Fig. 6, Fig. 7, and Fig. 8. A more detailed explanation of the results in this environment will be provided in Section 6.5.

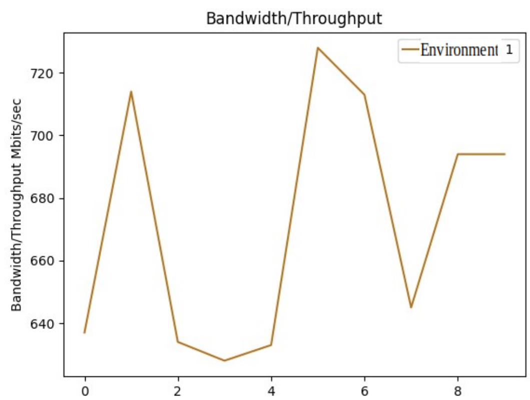

Figure 5: Throughput, local server test. The bandwidth values range from 634 Mbps to 728 Mbps.

This metric indicates the data transfer rate between the test points. The results show overall consistency in bandwidth, with minor variations between measurements.

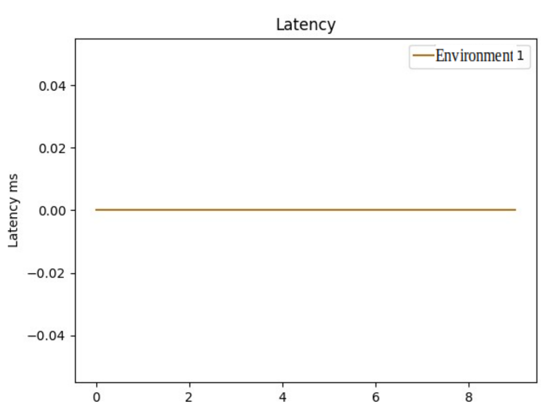

Figure 6: Latency tests were conducted on a local server. Latency value 0.0012 ms, close to zero (e.g., ≅ 0).

Therefore, there was no latency fluctuation. The result of latency was very low, close to zero. This indicates that there is no noticeable delay in data transmission.

This latency result is a good indication of a responsive network. End-to-end delay is a crucial metric and consists of four delays: transmission delay, propagation delay, queuing delay, and processing delay. Moreover, high communication delays are the main bottleneck in remote work, *home office*.

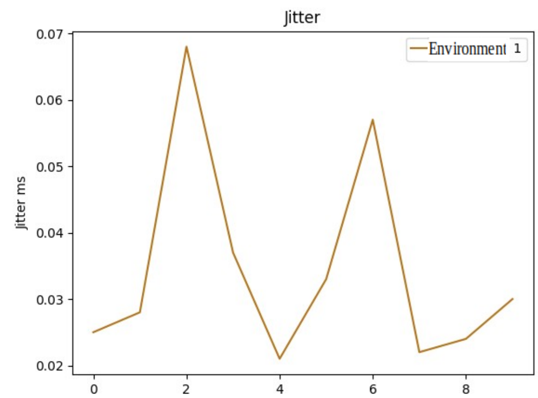

Figure 7: Jitter, local server test. The jitter values range from 0.021 ms to 0.068 ms.

Jitter measures the variation in packet delay within the network. Smaller jitter values indicate a more stable data transmission. In this case, the jitter values fall within an acceptable range, indicating a stable and consistent transmission.

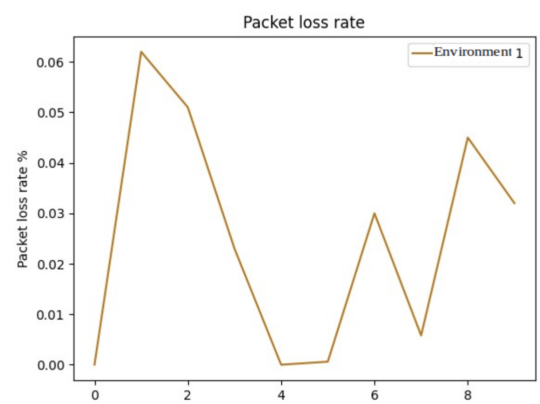

Figure 8: Packet loss rate, shown in percentage, local network test. The packet loss values range from 0% to 0.062%.

Packet loss indicates the percentage of packets that were not successfully delivered in the network. Lower values are desirable as they indicate a stable network.

6.2 Results of Environment 1 tests on public server

In this subsection, the graphs of the results for each metric in Environment 1 with a public server will be presented.

The individual results for each metric can be observed in Fig. 9, Fig. 10, Fig. 11, and Fig. 12. A more detailed explanation of the results in this environment will be provided in Section 6.6.

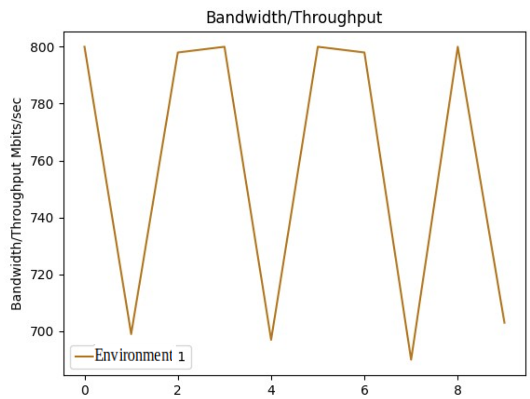

Figure 9: Throughput, public server test. The bandwidth values range from 697 Mbps to 800 Mbps.

This metric indicates the data transfer rate between the test points. The results show relatively consistent bandwidth values, with minor variations between measurements.

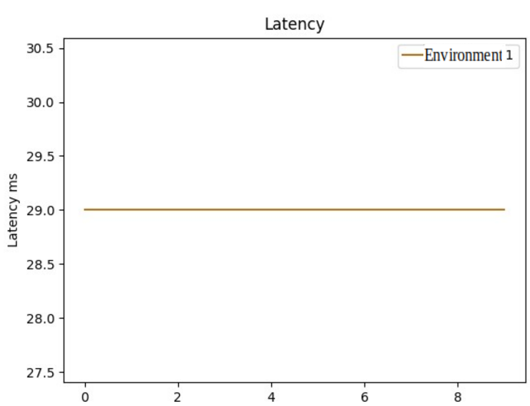

Figure 10: Latency, public server test. All latency values are equal to 29 ms.

Unlike the test on the local server, Environment 1, the test on the remote server had a delay of 29 ms. This is because the propagation delay is calculated as a function of the distance between the source and the destination. For example, the propagation delay is as follows: $PD = \frac{D}{PS}$. PD is the Propagation Delay, D is the distance between the source and the destination, and PS is the propagation speed of the considered medium (e.g., $2 * 10^8 ms$).

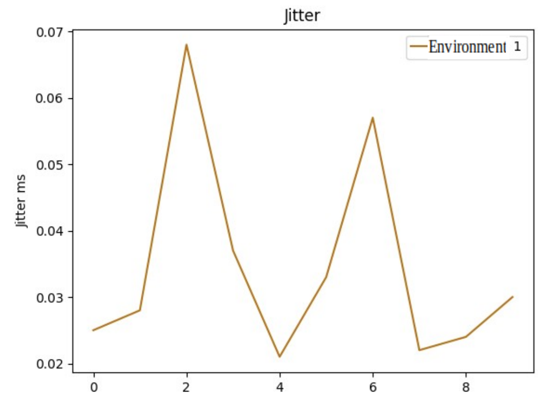

Figure 11: Jitter, test on the public server. The jitter values range from 0.145 ms to 0.429 ms.

Jitter measures the variation in packet delay within the network. These values, especially the maximum value of 0.429 ms, indicate a significant variation in packet delay.

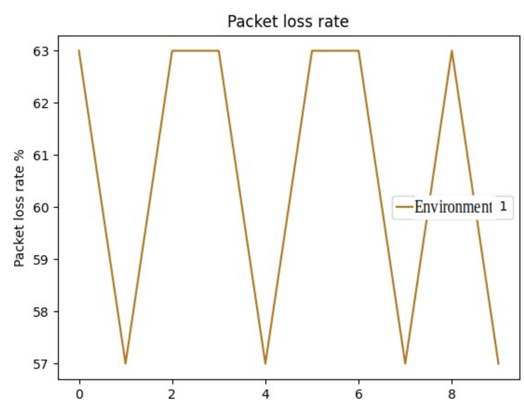

Figure 12: Packet loss rate, test on the public server. The packet loss values range from 57% to 63%.

Packet loss indicates the percentage of packets that were not successfully delivered in the network.

6.3 Results of Environment 2 tests on the local server

In this subsection, the graphs of the results for each metric in Environment 2 with a local server will be displayed. The individual results for each metric can be observed in Fig. 13, Fig. 14, Fig. 15, and Fig. 16. A more detailed explanation of the results in this environment will be provided in Section 6.5.

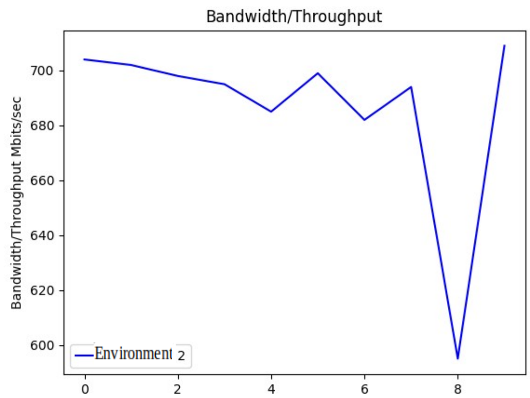

Figure 13: Throughput test on the local server. The bandwidth values range from 595 Mbps to 709 Mbps.

The results show considerable variation in bandwidth values, suggesting that the data transmission capacity in the network may be unstable.

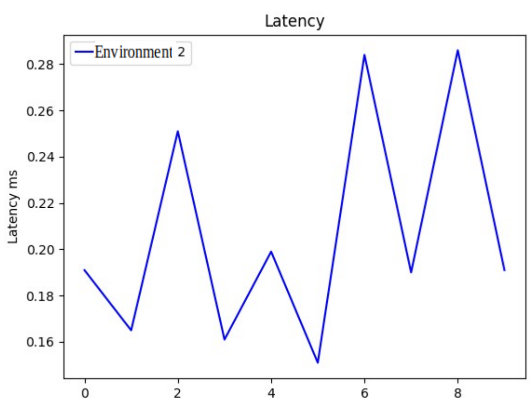

Figure 14: Latency, test on the local server. The latency values range from 0.151 ms to 0.284 ms.

Latency is the time it takes for packets to traverse the network. Consequently, according to the *International Telecommunication Union Sector Telecommunication* G.114 recommendation[1], delays below 0.150s are acceptable for most multimedia applications, and delays above 0.400s are impractical.

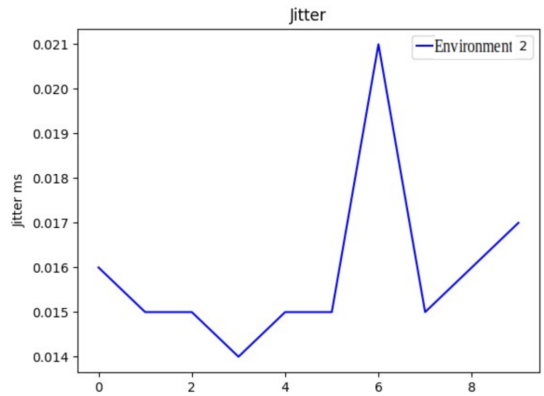

Figure 15: Jitter, test on the local server. The jitter values range from 0.014 ms to 0.021 ms.

Jitter measures the variation in packet delay within the network. These values fall within an acceptable range, indicating relatively stable transmission.

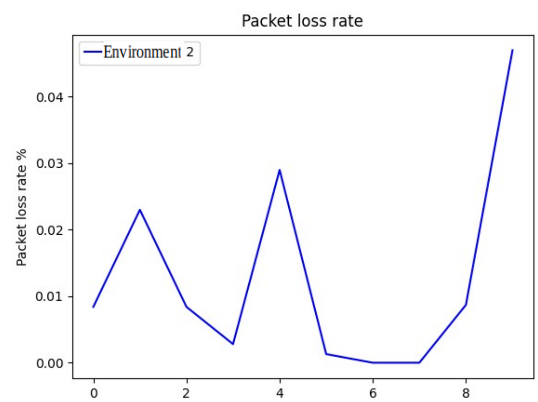

Figure 16: Packet loss rate, local server test. The packet loss values range from 0% to 0.047%.

Packet loss indicates the percentage of packets that were not successfully delivered in the network. Lower values are desirable for a more stable network.

6.4 Results of Environment 2 tests on public server

In this subsection, the graphs of the results for each metric in Environment 2 with a public server will be displayed.

The individual results for each metric can be observed in Fig. 17, Fig. 18, Fig. 19, and Fig. 20. A more detailed explanation of the results in this environment will be provided in Section 6.6.

[1] https://www.itu.int/rec/T-REC-G.114.

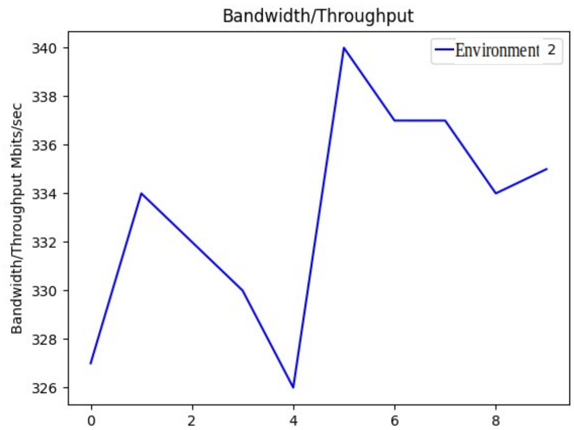

Figure 17: Throughput, public server test. The bandwidth values range from 326 Mbps to 340 Mbps. The results show relatively consistent bandwidth values, with minor variations between measurements.

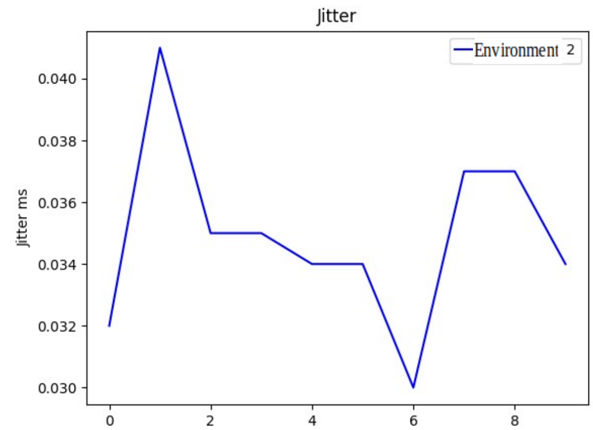

Figure 19: Jitter, public server test. The jitter values range from 0.030 ms to 0.041 ms. Jitter measures the variation in packet delay within the network.

Despite creating connections in iperf3 with a transmission rate of 800Mbps on the remote server (e.g., speedtest.sao1.edgoo.net), it was not possible to achieve values close to the maximum configured in the local environment.

This happens because, on the Internet, packets of the same data segment follow different paths, with different congestion and delays. Because of this, the transmission rate can be reduced.

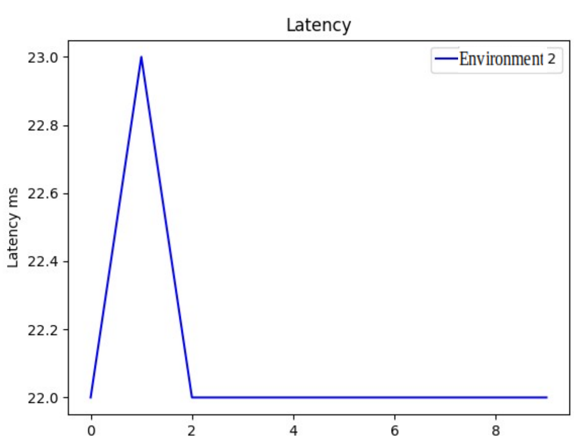

Figure 18: Latency, public server test. All latency values are between 22 ms and 23 ms. Latency is the time it takes for packets to traverse the network.

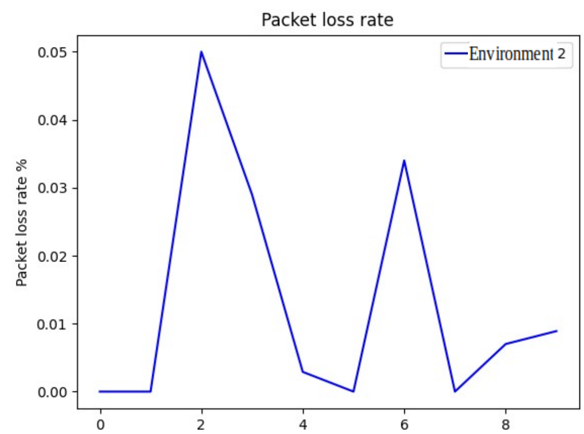

Figure 20: Packet loss rate, public server test. The packet loss values range from 0% to 0.05%. Packet loss indicates the percentage of packets that were not successfully delivered in the network.

6.5 Comparison between the tested environments in a local server

A comparative analysis of the test results was conducted to evaluate the difference in QoS between real machines and virtual machines in a local network context. In this subsection, we present this comparison. Thus, the graphical comparison of the metrics can be seen in Fig. 21, Fig. 22, Fig. 23, and Fig. 24.

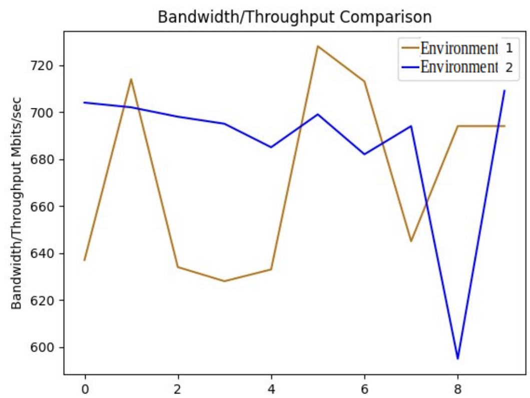

Figure 21: Throughput comparison, local server test.

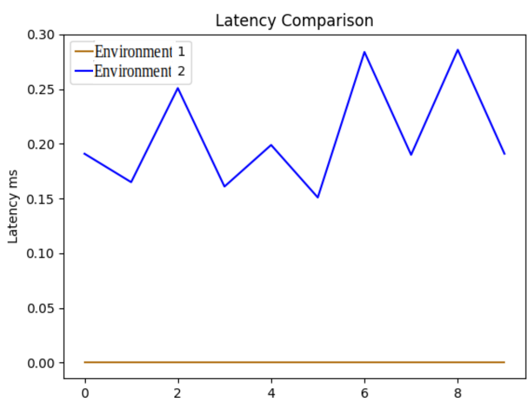

Figure 22: Latency comparison, local server test.

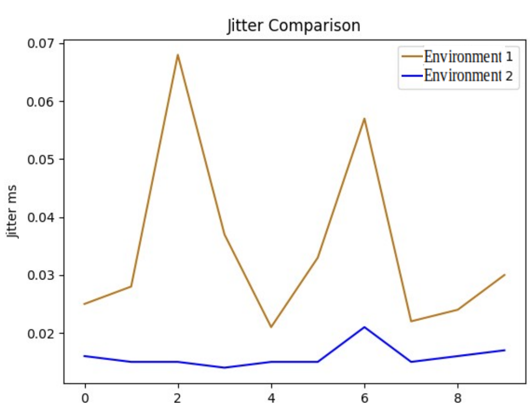

Figure 23: Jitter comparison, local server test.

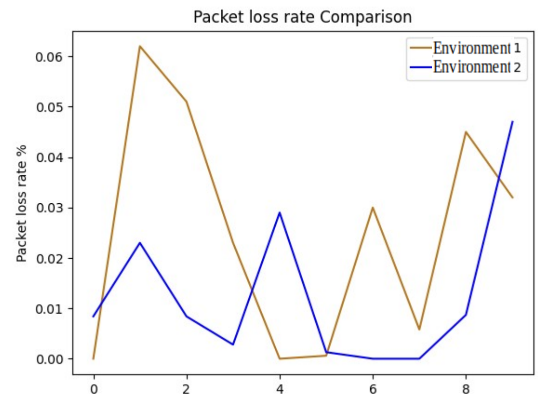

Figure 24: Packet loss rate comparison, local server test.

In Environment 1, the tests were conducted with the following average results:

- Bandwidth/Throughput: 637 Mbits/sec
- Jitter: 0.025 ms
- Packet loss rate: 0.962%
- Latency: 0.0012 ms (e.g., ≅ 0)

On the other hand, in Environment 2, the tests yielded the following average results:

- Bandwidth/Throughput: 704 Mbits/sec
- Jitter: 0.207 ms
- Packet loss rate: 0.158%
- Latency: 0.259 ms

A difference in jitter is observed between the two environments, with average values of 0.025 ms in Environment 1 and 0.207 ms in Environment 2. Additionally, Environment 1 exhibited an average packet loss rate of 0.962%, whereas Environment 2 recorded an average packet loss rate of 0.158%. As for latency, Environment 1 showed no significant latency, while Environment 2 had an average of 0.259 ms.

These results indicate that Environment 2, composed of virtual machines, may experience higher jitter variation and lower packet loss than Environment 1, consisting of physical machines. However, it is essential to note that both environments achieved satisfactory performance in terms of bandwidth and latency.

According to Tanenbaum's stringent quality of service requirements, certain services such as email, file transfer, web access, and remote login require network reliability, as packet loss can negatively impact their effectiveness. Additionally, video conferencing is highly sensitive to three QoS metrics: jitter, latency, and bandwidth (or throughput). Services such as web browsing and remote login are highly affected by latency. On the other hand, real-time services like telephony and video conferencing require low latency to function correctly. Users will consider the connection unacceptable if there is a constant delay of 2,000 milliseconds in every word during a phone call. Conversely, video, especially audio, is susceptible to jitter. The result will be detrimental if the transmission time varies randomly between 1 and 2 seconds (Tanenbaum and Wetherall, 2011).

Therefore, in Environment 1, services such as email, file transfer, web access, and remote login can be considered reliable, as the packet loss rate is relatively low, at just 0.962%. Additionally, the latency is almost zero, which is favorable for Internet browsing and remote login services sensitive to this aspect. However, the results show that the jitter is extremely low, with only 0.025 ms, which is positive for video and audio services that are highly sensitive to this metric. As for throughput (bandwidth), an average value of 637 Mbits/sec was recorded.

In Environment 2, the average results indicate slightly better performance compared to Environment 1. The throughput reached an average of 704 Mbits/sec, demonstrating increased bandwidth. However, the jitter showed an average value of 0.207 ms, which can negatively impact services sensitive to temporal variations, such as video and audio. The packet loss rate is also relatively low, at 0.158%. Regarding latency, an average value of 0.259 ms was observed.

Both environments perform adequately for services such as email, file transfer, web access, and remote login due to the low packet loss rate and almost negligible latency. However, Environment 2 exhibits an improvement in throughput, which can benefit services requiring higher bandwidth.

Both environments show satisfactory results with low jitter values for jitter-sensitive services like video and audio. However, it is essential to note that Environment 1 recorded even lower jitter than Environment 2.

6.6 Comparison between the tested environments on a public server

This subsection presents the comparative analysis of the tests conducted on the public server. The graphical comparison of the metrics can be observed in Fig. 25, Fig. 26, Fig. 27, and Fig. 28.

It is important to note that although Environment 2 shows worse performance in the throughput metric, this does not necessarily imply that the use of virtual machines is the cause. A more likely explanation is that, when running iperf3 in this environment, packets may have followed routes with varying bandwidths or congested paths. As a result, these factors could negatively impact certain performance metrics, such as throughput. Therefore, when analyzing network traffic flows, it's essential to consider all these variables.

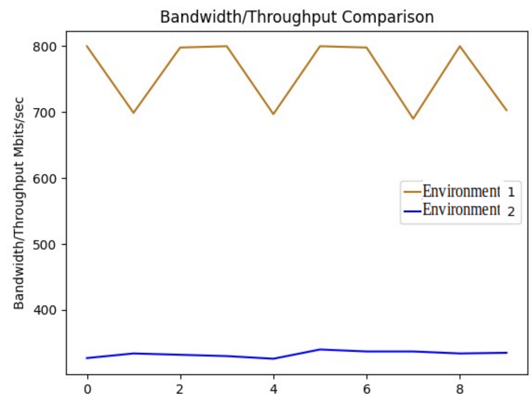

Figure 25: Throughput test on the public server.

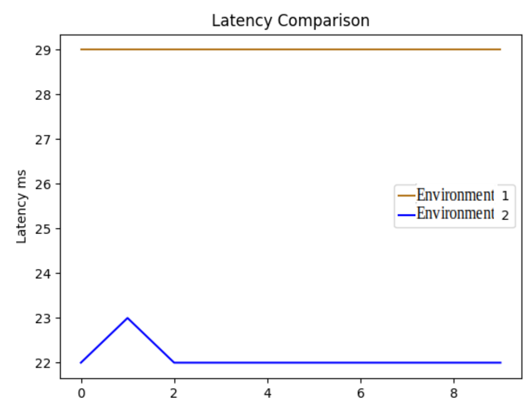

Figure 26: Latency, test on the public server.

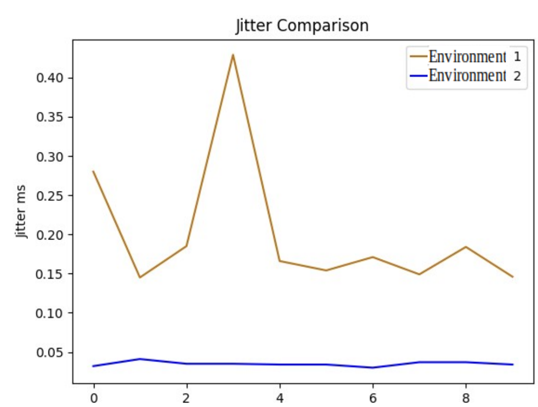

Figure 27: Jitter, test on the public server.

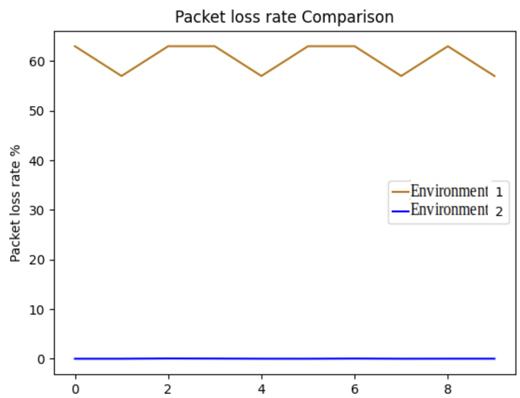

Figure 28: Packet loss rate, public server test.

In Environment 1, the tests on the public server yielded the following average results:

- Bandwidth/Throughput: 799 Mbits/sec
- Jitter: 0.231 ms
- Packet loss rate: 60.3%
- Latency: 29 ms

On the other hand, in Environment 2, the tests on the public server produced the following average results:

- Bandwidth/Throughput: 333 Mbits/sec
- Jitter: 0.035 ms
- Packet loss rate: 0.014%
- Latency: 22 ms

In this context, it is possible to observe a significant difference in the results between the two environments. Environment 1 recorded average values of 799 Mbits/sec for bandwidth, 0.231 ms for jitter, a packet loss rate of 60.3%, and an average latency of 29 ms. In Environment 2, the average results were 333 Mbits/sec for bandwidth, 0.035 ms for jitter, a packet loss rate of 0.014%, and an average latency of 22 ms.

These results demonstrate that Environment 1 exhibited a higher packet loss rate than Environment 2, while Environment 2 achieved a lower jitter and slightly lower latency.

Environment 1's throughput is high, reaching 799 Mbits/sec, which is positive. However, the jitter is at 0.231 ms, which can affect video and audio quality in a video conference, as jitter refers to the variation in packet delay. Additionally, the packet loss rate is high, reaching 60.3%, which can negatively impact email, file transfer, and web access services. The latency of 29 ms can also be considered relatively high for latency-sensitive services like web browsing and remote login.

On the other hand, in Environment 2, the throughput is lower compared to Environment 1, reaching 333 Mbits/sec. However, the jitter is at a low value of 0.035 ms, which is positive for ensuring audio and video quality in a video conference. The packet loss rate is also typical, at 0.014%, which benefits services that require reliability, such as email and file transfer. The latency of 22 ms is relatively low, which is favorable for latency-sensitive services like web browsing and remote login.

It is essential to highlight that the tests on the public server were conducted under similar conditions in both environments, ensuring the validity of the comparison. Furthermore, these tests provide an additional perspective on the performance of the environments in a broader context.

6.7 General Comments

After analyzing test Environments 1 and 2, each metric's usability sensitivity points were observed, considering Tanenbaum's QoS requirements rigidity standards (Tanenbaum and Wetherall, 2011). Based on the results obtained in each environment, it is possible to evaluate performance in different types of usability, such as remote access to another machine, video calls, or data transfer over the network. Consequently, only the results from the tested environments on the public server are discussed to provide a more realistic view of the data. Furthermore, it allows us to understand how each environment would perform in certain services that require high reliability. For example, it implies minimizing packet loss to ensure efficient communication. Additionally, some on-demand services require minimal variation in delivery time (jitter) to provide a satisfactory user experience.

From the *bandwidth/throughput* results, Environment 1 exhibits higher transmission capacity than Environment 2. However, both environments had relatively low values regarding jitter, indicating good stability in on-demand audio and video transmission. In terms of packet loss rate, Environment 2 performs significantly better. It means high reliability in data transmission, especially for packet loss-sensitive services such as email sending and file transfer. As for latency, there are no significant differences between the two environments, with both maintaining a constant value between 22ms and 29ms, which is suitable for most remote access applications, video calls, and web browsing, following Tanenbaum's QoS requirements rigidity standards (Tanenbaum and Wetherall, 2011).

Furthermore, when comparing our results with those presented in the studies cited in Section 2 (Related Work), we highlight that our research evaluates *QoS* using the *Iperf3* tool, based on the metrics of *Throughput*, *End-to-End Delay*, *Jitter*, and *Packet Loss Rate (PLR)*, across environments comprising real and virtual machines, under two distinct network topologies: LAN and WAN. In this context, we selected the *UDP* protocol due to its low end-to-end delay and high responsiveness, features well-suited for streaming applications. For instance, the study in (Reisslein et al., 1999) reported low delay values using the *Real-Time Streaming Protocol (RTSP)*, which relies on UDP in a conventional scenario. In comparison, our results for end-to-end delay, *Jitter*, and PLR outperformed those in (Reisslein et al., 1999). Consequently, when comparing our findings with the works in Section 2 (Related Work), we observe that these studies generally aim to optimize *QoS* parameters by maximizing throughput and minimizing end-to-end delay, *Jitter*, and PLR.

Thus, key distinctions lie in the testing environments and transport protocols employed, which vary according to application contexts—such as IoT (Khamosh et al., 2023), QoE (Wang et al., 2014), SDN (Facchini et al., 2020), Security and VANETs (Nazia Tabassum, 2022), and Video Streaming (Facchini et al., 2020). Another important aspect is the variety of experimental setups used, including simulations with NS-2, OPNET, ODL, and physical testbeds. Thus, the diversity in technologies, simulators, real-world environments, applications, and communication protocols led to varying results, each considered optimal within the specific contexts in which they were evaluated.

In general, networks are expected to deliver high throughput and low latency. However, achieving this in real-world scenarios is challenging due to various adverse conditions that can degrade communication performance.

7 Conclusions

This study presented the concept of QoS and an experiment comparing the QoS of traffic generated between virtual and physical machines. In this context, one of the main contributions of this study was to present QoS as a technology and a means to evaluate the quality and reliability of a network, highlighting its metrics and utilizing the Iperf3 tool to generate results for analysis. A significant aspect of this work was the design of the test environments, referred to as Environments 1 and 2, where efforts were made to minimize interferences and simulate a neutral data traffic pattern in each environment. Furthermore, it allowed for the framing of different commonly used use cases during the pandemic (Lee et al., 2021) when there was a significant increase in Internet usage (OECD, 2020).

Consequently, the main performance differences among the proposed environments were presented based on QoS metrics such as throughput, end-to-end delay, jitter, and packet loss. Understanding these results is crucial to support future decisions regarding network deployment and resource scaling, enabling a more efficient allocation of available resources in each use case.

As future work, it would be interesting to evaluate QoS metrics in other environments, such as cloud environments or using containers (Docker), since the Iperf3 tool already supports such testing. Additionally, a comparison could be made with a simulated environment, such as NS-2. These additional investigations can provide a more comprehensive insight into performance and quality in different network deployment contexts.

Acknowledgments

We gratefully acknowledge the Faculty of Exact Sciences and Technology (FACET) at the State University of Mato Grosso (UNEMAT) for providing the technological resources of the Computer Networks Laboratory, which enabled us to conduct tests using the Iperf server: speedtest.sa01.edgoo.net.

Author Contributions

All authors have contributed equally to the development of this work.

References

Agusriandi, A. and Elihami, E. (2020). Developing delay jitter, throughput, and package lost iperf3 for learning islamic education, *Jutkel: Jurnal Telekomunikasi, Kendali dan Listrik* **2**(1): 23–30. Available at https://web.archive.org/web/20210814174626/https://ummaspul.e-journal.id/Jutkel/article/download/602/332.

Community, I. S. L. (2024). Public iperf3 servers: A list of public iperf3 servers. Available at https://iperf3serverlist.net/.

da Costa, G. H. (2008). Métricas para avaliação de desempenho em redes qos sobre ip. Available at https://lume.ufrgs.br/handle/10183/15972.

Dhafer R. Zaghar, T. S. A. A. A. W. (2013). Simplified the qos factor for the ad-hoc network using fuzzy technique, *Int'l J. of Communications, Network and System Sciences* **6**: 381–387. https://doi.org/10.4236/IJCNS.2013.69041.

Duc, C. T., Rosdiazli, I., Sagayan, A. V., Nordin, S. and Miya, H. S. (2016). Effect of network induced delays on wirelesshart control system, *2016 6th International Conference on Intelligent and Advanced Systems (ICIAS)*, IEEE, pp. 1–5. https://doi.org/10.1109/ICIAS.2016.7824045.

Dugan, J., Elliott, S., Mah, B. A., Poskanzer, J. and Prabhu, K. (2024). Iperf - the ultimate speed test tool for tcp, udp, and sctp. test the limits of your network and internet neutrality test. Available at https://iperf.fr/.

Facchini, Alberto, H., Pérez, Cristóbal, S., Alejandro, D. and Fabián, H. (2020). Evaluation of video traffic behavior metrics in an experimental multicast network, *Enfoque UTE* **11**(1): 15–27. https://doi.org/10.29019/enfoque.v11n1.576.

Forouzan, B. A. (2007). *Data Communications and Networking*, 4 edn, McGraw-Hill High Education, Cupertino, CA.

Henrique, D. P. and Alves, J. N. (2014). Ferramenta iperf: geração e medição de tráfego tcp e udp, *Notas Técnicas* **4**(2). https://doi.org/10.7437/NT2236-7640/2014.02.003.

ISO/IEC13236 (1998). Information technology - quality of service - framework. Available at https://www.iso.org/standard/27993.html.

Kesavan, D., Periathambi, E. and Chockkalingam, A. (2023). A bipartite graph based proportional fair scheduling strategy to improve throughput with multiple resource blocks, *International Journal of Electrical and Computer Engineering (IJECE)* . http://doi.org/10.11591/ijece.v13i4.

Khamosh, A., Ahmadi, A. R., Hamdard, M. A. A. J. and Nasrat, S. A. N. (2023). Relationship between iot service user quality and network qos factors, *Journal for Research in Applied Sciences and Biotechnology* . https://doi.org/10.55544/jrasb.2.2.21.

Lee, R., Colleen, M. and Haley, N. (2021). The internet and the pandemic. Available at https://www.pewresearch.org/internet/2021/09/01/the-internet-and-the-pandemic/.

Mazhar, T., Malik, M. A., Mohsan, S. A. H., Li, Y., Haq, I., Ghorashi, S., Karim, F. K. and Mostafa, S. M. (2023). Quality of service (qos) performance analysis in a traffic engineering model for next-generation wireless sensor networks, *Symmetry* **15**(2). https://doi.org/10.3390/sym15020513.

Nazia Tabassum, C. R. (2022). Review on qos and security challenges associated with the internet of vehicles in cloud computing, *TRENDS on Internet of Things* . https://doi.org/10.1016/j.measen.2022.100562.

OECD (2020). Keeping the internet up and running in times of crisis. Available at https://read.oecd.org/10.1787/4017c4c9-en?format=pdf.

Pratama, A. A. and Wikantyasa, A. (2019). Implementasi dan analisis simulasi qos dan perfomance device dengan menggunakan onos dan iperf3, *Jurnal Informatika Universitas Pamulang* . http://dx.doi.org/10.32493/informatika.v4i2.2730.

Reisslein, M., Ross, K. W. and Rajagopal, S. (1999). Guaranteeing statistical qos to regulated traffic: the single node case, *IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)* **3**: 1061–1072. Available at http://cis.poly.edu/~ross/papers/GuaranteeQoSsingleNode.pdf.

Susom, A. A. (2018). Effectiveness of routing protocols for different networking scenarios, *Advances in Science, Technology and Engineering Systems Journal* . https://doi.org/10.25046/AJ030412.

Tanenbaum, A. S. and Wetherall, D. J. (2011). *Computer Networks*, 5 edn, Person, Seattle, WA.

Valencia, J. C. C., Muñoz, W. Y. C. and Golondrino, G. E. C. (2020). Análise de qos para iptv em um ambiente de redes definidas por software, *Revista Ingenierías Universidad de Medellín* **19**(36): 29–51. https://doi.org/10.22395/rium.v19n36a2.

Villarim, A. W. R., da Rocha Souto, C. and Villarim, M. R. (2023). Experimental analysis of lora communication in underground environments for iot applications, *Revista Brasileira de Computação Aplicada (RBCA)* **15**(3): 38–47. https://doi.org/10.5335/rbca.v15i3.14758.

Wang, Z., Li, L., Wang, W., Wan, Z., Fang, Y. and Cai, C. (2014). A study on qos/qoe correlation model in wireless-network, *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific* pp. 1–6. https://doi.org/10.1109/APSIPA.2014.7041760.