Machines and Algorithms
http://www.knovell.org/mna

Review Article

Comparative Analysis of TCP and UDP Transport Layer Protocols: A Computational Study

Perveen Akhtar[1*], Ismaeel Yousaf[2], and Aqsa Jameel[3]

[1]Department of Computer Science, Institute of Southern Punjab, Multan, Pakistan
[2]Department of Information Technology, Bahauddin Zakariya University, Multan, Pakistan
[3]Institute of Computer Science & Information Technology, The Women University, Multan, Pakistan
[*]Corresponding Author. Email: perveen.mitroo@gmail.com

Received: 12 April 2022; Revised: 03 June 2022; Accepted: 17 July 2022; Published: 17 August 2022
AID: 001-02-000006

Abstract: All applications use the www and HTTP protocol over internet that is responsible to handle all sorts of communication over the internet. It builds on TCP model, but TCP is not compatible with the shortest transmissions that consist of important part of web traffic. Application that runs with the TCP ensures the transparency of the ordered data without the loss of any packet but it will cause inefficiency and long delays in short conversations. In this research paper, the comparative analysis of both transport layer protocols TCP and UDP will be discussed. This research paper will also be focused on some main terminologies of TCP and UDP. It will also help us to know the key differences between them. In this paper we will also be discussed the performance of TCP protocol over wireless systems.

Keywords: IP Address, TCP Protocol, UDP Protocol, HTTP, Network Addresses, Internet.

1 Introduction

A computer network represents a complex infrastructure, intricately connecting a multitude of devices to facilitate seamless data and information sharing. These intricate networks predominantly rely on wired connections, where data takes the form of electromagnetic waves traversing diverse communication mediums. The evolution of computer networks has been a fascinating journey, from the early days of point-to-point connections to the sophisticated global networks we have today.

Within this intricate network ecosystem, each individual device assumes the role of a node. While the majority of nodes are computer systems, it is essential to note that the network also encompasses a range of devices, including dumb terminals, bridges, routers, and printers. Importantly, each node or device within the network boasts a unique address [1], akin to a numerical identifier. The length of these addresses' exhibits variability, with networks featuring longer addresses demonstrating the capacity to accommodate a larger number of simultaneously connected devices [2][3].

© 2022 The Authors. Published by Machines and Algorithms. This article is licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the work is properly cited.
https://doi.org/10.66108/mna.v1i2.20

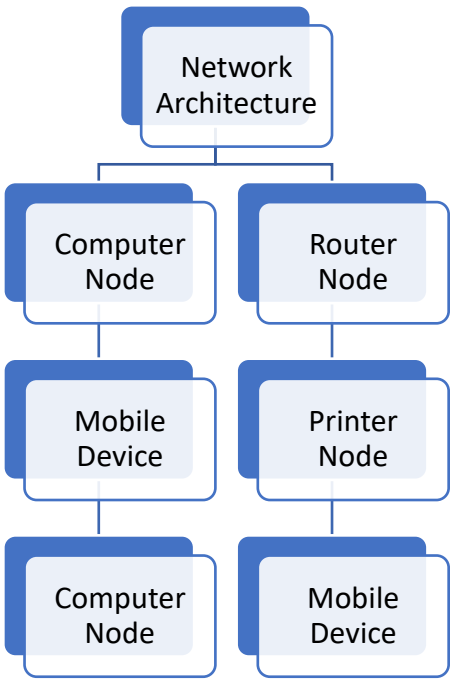

Figure 1: Illustration of Network Architecture and Device Types with Associated IP Addresses.

Address allocation methodologies exhibit variations across different networks. For instance, in the domain of AppleTalk, hosts are assigned addresses randomly, ensuring that no two devices share the same address. In stark contrast, Ethernet addresses are intrinsically linked to the physical hardware.

In most cases, the responsibility of assigning internet addresses to individual computers falls upon organizations. However, it is crucial to acknowledge that these organizations acquire address assignments from Internet Service Providers (ISPs). It is common practice for nodes within diverse networks to be equipped with human-readable names to facilitate effortless identification. These names may undergo alterations without impacting the underlying addresses, or conversely, addresses may undergo changes while the names remain unaltered [4].

Furthermore, it is noteworthy that a single address may be associated with multiple names, and conversely, a solitary name may correspond to several distinct addresses. Contemporary networks overwhelmingly adhere to a packet-switched architecture, whereby data is systematically divided into discrete packets for transmission. Each packet is treated autonomously, with multiple packets from varied networks coexisting on the same transmission medium without encountering interference. Packets also offer the added advantage of enabling the verification of data integrity through the utilization of checksums [5]. The sphere of data communication within computers is diligently governed by protocols, which stand as comprehensive sets of rules and principles dictating the nuances of how computers engage with other devices. These protocols meticulously delineate address formats, data packetization processes, and other pivotal facets of the data exchange process [6].

In the contemporary technological landscape, there exists a burgeoning demand for wireless technologies such as GPRS, WIFI, and WiMAX. Simultaneously, mobile multimedia applications have experienced a surge in popularity, obligating them to meet rigorous quality requirements while judiciously optimizing network resources [7].

The transport layer emerges as a central determinant in satisfying these multifaceted demands. For a thorough comprehension of network architecture, it becomes imperative to scrutinize various transport layer protocols and evaluate their competence in delivering high-quality services. This research paper casts its focus on two primary transport layer protocols: the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP). TCP offers a dependable method for data transportation, albeit potentially

incurring delays [8].

In contrast, UDP presents the advantage of lower latency but sacrifices congestion control mechanisms. In the contemporary network environment, Quality of Service (QoS) stands as a paramount concern [9][10], and transport layer protocols wield a substantial influence on QoS outcomes. Due to its low-latency characteristics, UDP commonly finds application in multimedia contexts. Nevertheless, it is critical to acknowledge that the QoS requisites of specific multimedia applications may exhibit variance, potentially rendering UDP insufficient for yielding satisfactory results. By explicitly elucidating these key points, we aim to provide a clearer foundation for understanding the research's context and significance.

2 Related Work

2.1 Network Layer

Transportation of data across network is considered as a complex function. We need to be carefully tuned it with the physical as well as logical features of data which is going to be transmitted. Program or set of instructions that are required to transmit data across different networks should determine how to route the packets among different nodes, detection and error correction, how to change digital signals into analogue signals and how to prevent from collision [11]. Network communication will be divided into different layers in order to hide this complexity from end users or developers [12]. Different levels of abstraction are given at every layer. There are four layers in TCP/IP network.

2.2 Transport Layer

Transport layer supports end-to-end communication between different nodes within the network. It is considered as the main component of the protocol hierarchy. It also helps to support cost effective and reliable services of transmission from sender to receiver. Transport layer provides the following services [13]:

- Flow and error control
- Reliable delivery of packets
- Connection oriented services and applications
- Multiplexing

Lowest layer of the protocol stack is considered as the physical layer. This layer is only responsible for transmitting bits on the medium. Data link layer is the above layer of physical layer which is responsible to provide a link between several nodes in order to deliver a data from sender to receiver. It is also responsible to provide a reliable delivery of data, error detection, framing of bits and media access control [14]. Network layer is the above layer of data link layer. It is responsible to transfer the packets across the network. The next layer is the transport layer.

The main function of this layer is to provide the data transport services between two devices in the network [15]. De multiplexing of applications is also performed at this layer. The transport layer does not have its own address, so it will have to use the network layer addresses in order to recognize a connection. Application layer is considered as the upper layer of protocol stack that is identified by a name. It is only visible to users to identify the node [16]. There are seven layers that are working in the protocol stack. Every layer has a particular function and a set of protocols and performs specific services. Protocols are also performing services in each layer. This paper will be focused on the transport layer and protocols that are working on this layer. The transport layer has various protocols but we will be discussed about its two basic protocols known as TCP protocol and UDP protocol.

2.2.1 Transport Layer Services

Following are the Services of transport layer protocol:

- Connection-oriented services: It is going to determine the connection as a stream of bits gives different advantages to applications. It is easy to deal with the connection-oriented architectures as compared to connection less architecture.

- Byte-oriented processing: It will easy to handle the data as a sequence of bytes instead of handling the messages in the previous format of communication system. It will help the applications to work with different formats of messages.
- In order delivery: It the most desirable factor that packets must receive in the same order as they were sent. It is only possible through the segment number that receiver transmitting to the process in the same order.
- Reliable delivery: The packets can be lost due to the problems of network traffic deadlock and errors during the transmission. There is a mechanism known as checksum that helps to detect whether the packet is corrupted or not. When receiver receives the packet correctly, it will send an acknowledgment (ACK) for the verification by sending NACK to the sender.
- Flow control: If the source sends data at higher transmitting rate, it will not be controlled by the destination and causing receiver buffer overrun [17]. So that the data transmission rate should be managed between the two nodes in the network.
- Congestion control: Congestion control mechanism handles the traffic in order to prevent congestion in the network.
- Services of Multiplexing: A port has many endpoints within the one device in the network. Each computer application will handle the information at its own port. It helps to use more than one network services simultaneously.

2.2.2 Transport Layer Protocols

In protocol hierarchy, transport layer is considered as an important part that facilitates end-to-end transmission between two systems in the network [18]. There are many services which are provided by the transport layer protocols such as a reliable transmission, flow control, congestion control and in-sequence delivery. All of these services provide the quality of communication in the network. The right choice of transport layer protocols may affect the quality of requirements in case of multimedia applications such as less delay, throughput and packet loss etc. [19].

Protocols of transport layer are given below:

- TCP Protocol (Transmission control protocol): In internet protocol suite, TCP is considered as the basic protocol. It is responsible to provide quality of services such as reliable transmission, flow control, congestion control and in-sequence delivery [20].
- UDP Protocol (User datagram protocol): In internet protocol suite, UDP is also considered as the basic protocol. Sender transmits the message to the receiver in the form of datagram with the help of UDP in the network without setting up a channel of transmission before the real communication is begun [21].
- DCCP Protocol (Datagram congestion control protocol): There is another transport layer protocol known as DCCP protocol which is considered as the message oriented. It will provide various services like congestion control etc. [22].
- UDP protocol: UDP protocol is responsible to send packets in the form of datagram across the network.
- SCTP Protocol (Stream control transmission protocol): SCTP is also a transport layer protocol. Its function is same as TCP protocol and UDP protocol. It also gives similar services like both of these protocols. SCTP is considered as message oriented like UDP and it is also responsible to provide the ordered delivery of packets like TCP protocol [23].
- RSVP (Resource reservation protocol): It is resource reservation protocol that is responsible to reserves the resources in the network for various integrated services. It also gives the infrastructure of resource reservation that is started at the source for multicasting or unicasting [24].
- RIP Protocol (Routing information protocol): It is the routing information protocol that

uses the hop count like metric in order to rout the packets [25]. It is also applied limit on the hop numbers that are allowed in a way from sender to receiver in order to avoid routing loops [26].

TCP is considered as a slandered protocol that is more suitable for wired networks and not suitable in case of wireless networks. UDP is used by some multimedia applications for transmission. UDP is considered as the more appropriate protocol for time constraint's applications instead of ordered delivery. On the other hand, TCP is considered as more appropriate protocol for those applications which are not depend on time constraint and reliable delivery of packets is very important.

2.2.3 UDP OR User Datagram Protocol

UDP is considered as the simple transport layer protocol that does not make sure to provide any reliable transmission and ordered delivery of message [27]. Multicasting and broadcasting services are provided by UDP. In this case, it is important to deliver messages within the specific amount of time duration instead of reliable transmission of messages. Packets of UDP are known as datagram that has two parts. These two parts are header and a payload. UDP implements (CRC) cyclic redundancy check. It will check the integrity of data in the network.

So, it will be able to detect errors in the packet. When it will detect an error in the message than it will discard that packet and packet may be lost. This protocol is responsible to the availability of the datagram for packet-switched communication in the network devices. It does not support any congestion control method, so it becomes a major issue today. Some researchers have introduced new methods in UDP to perform adaptive congestion control but this issue still exists [28].

Basic functionality of UDP: UDP protocol has very simple operation. When an application layer is invoked UDP, it will perform the following operations [29]:

- UDP converts the data into datagram for users.
- It will transmit these data grams to the IP layer.

From IP layer, these data grams then sent to UDP. UDP protocol will delete the message within the datagram and send it to an application layer. Port will recognize the application that is using the services of UDP. It is supposed to be an address of the application. At the destination side, the port number is using by the UDP client, so it will be able to know which application is forwarded with user data [30].

Applications of User Data gram Protocol (UDP): UDP is used by different applications as their transport layer protocol such as protocol of dynamic host configuration, protocol of simple network management and protocol of routing information etc. [31]. Video and voice traffic is usually sent through UDP protocol over the network. TCP and UDP protocol will perform functions simultaneously in the network.

2.3 TCP Protocol OR Transmission Control Protocol

TCP is a connection-oriented protocol and it is responsible to provide a reliable transmission. TCP provides method of positive acknowledgments and method of a congestion control that manages the transmission rate in the network when it is overloaded [32]. TCP offers a variety of services such as congestion control service, reliability and flow control etc.

TCP is also considered as robust protocol because we can take this protocol in different situations of network. According to the principle, TCP must operate over a wide range of communication networks like circuit-switched networks or from wired connection to the packet-switched networks [33]. In order to provide a reliable transmission of data across network, TCP provides a method where the sender handles with buffer, known as sliding window that is being transmitted by the receiver [34]. When receiver receives the packet, it will send acknowledgment (ACK) packet to the sender.

When the sender gets an ACK packet from the receiver in its window, it will delete that packet until it will have to transmit the data to the receiver successfully. This procedure is known as transmission that is window based. TCP will implement this procedure to flow control, so a destination can inform to source

when it will not be able to operate the data at the rate in which it is reaching at destination [35]. This procedure also helps the sender to prevent the buffer overflow.

2.3.1 Functions of Transmission Control Protocol (TCP)

Data transfer: Providing a data continuously between the users within the network through segments is the major responsibility of TCP.

- Reliable delivery: TCP can also recover the data which may be lost over the network. A sequence number will be assigned to each segment which is transferred over the network and getting a positive ACK when message will be delivered successfully [36]. If it will not receive ACK during a fixed interval of time, the data will be required to retransmit. Segment can be out of order. So, by using the sequence number, the receiver can order the segment in right direction to prevent the duplicate messages. TCP can also implement checksum to each segment in order to handle the damaged or lost messages. After checking will be completed at the receiver side, the lost segments will be discarded.
- Flow control: TCP offers a method which helps the receiver to control the data that is sent by the source. The procedure of returning a "window" along with the ACK packet will make it possible. It will help to provide a reliable transmission across network [37].
- Multiplexing: TCP also offers a set of ports within every host, so various processes can be used facilities of TCP network communication within a single host simultaneously. It will form a socket when it is concatenated with the host and network addresses. The pair of sockets identifies every connection uniquely. Thus, different connections can use a pair of sockets simultaneously.
- Connections: Window sizes, sequence numbers and pair of sockets can make a connection.

A pair of sockets is used to identify a connection uniquely. When the two processes start communication, their TCPs would make a connection first. i.e., initialized the status information at both sides [38]. when the communication will be completed, connection will be closed.

2.3.2 TCP Segment

By comparing a UDP datagram with a TCP segment, there is a huge difference between these protocols. UDP is faster than TCP but it does not provide a reliable transmission [39].

Following a brief description of the TCP segment format and its each field:

- Address of Source port: It will identify the segment of the source.
- Address of Destination port: Sender is used this address to transmit the segment to the destination. It is 16-bit destination port number.
- Sequence number: It is 32-bit in the segment.
- Acknowledgment number: It is 32-bit number. Value of the next sequence number of the segment will be identified. When the connection is established, it is required to transmit.
- Reserved: It is of 6 bits and it must be kept zero and is reserved for further use.
- Control bits: There are 6 control bits such as FIN, SYN, RST, PSH, ACK and URG.
- Window: It is of 16 bits that identifies the data octet number starting with one identified in the field of acknowledgment that the source of the segment needs to be received.
- Checksum: It is also 16-bit number.
- Urgent pointer: This field offers communication to the present value of urgent pointer.
- Option: It possesses the space at the end of the header. Options are also present in checksum.

2.3.3 Applications of TCP

Following are the applications of TCP:

- File transfer protocol (FTP): FTP offers a method in order to transmit the data files between system devices. All types of web browsers and client-server applications use TCP protocol.
- Hypertext transfer protocol (HTTP): HTTP is considered as the most popular protocol across the internet. It makes web pages to move over the network.
- Interactive mail access protocol (IMAP): It facilitates the client to handle the mailboxes. He can also access e-mail messages over the network through this protocol.
- Post office protocol (POP): It is used in e-mail applications. E-mails that are placed on a remote server can be read and deleted by the client through this protocol.
- Remote login (Rlogin): It will provide login facilities over the network.
- Simple mail transfer protocol (SMTP): Its main feature is to provide system to system delivery.
- Secure shell (SSH): It will provide encryption of data and provide remote access.

2.3.4 Transmission Control Protocol (TCP) in Wireless Networks

Wired networks use TCP through links with stationary hosts and have some bit error rates. A lost packet is always considered as the loss as according to the TCP. It is just because of network congestion [7]. This hypothesis may right for the wired network but it cannot suit to wireless networks. Many aspects may contribute to the loss of packets within wireless network such as frequent disconnections and high error rate. We are required to take different measures in case of packet loss in TCP in order to enhance the performance within wireless networks. There are some mechanisms that deal with the performance issue of TCP protocol [40].

The approach of split connection: We can break TCP connection into two different connections. We will establish one connection between the base station and fixed host and other connection between the mobile host and base station. If we will break the TCP connection into two different connections, there are some specific protocols that are designed especially for wireless network between the base station and mobile host [41].

The fast-retransmit approach: Loss of packets and delays will occur in case of moving the mobile hosts and the base station handled the cellular hand-offs [5]. As we know that TCP offers a congestion control method, so these losses will be considered as damages due to the problem of network congestion. In many situations, it is necessary to have a time out in order to restart the flow of packets among hosts. If we are going to wait of these timeouts during cell switching, long delays can occur. The goal of this methodology is to minimize these delays. It is only possible if sender is trying to resend the segments through artificially triggering the fast resend method of TCP [42]. It will be completed by transmitting three duplicate ACKs after the completion of cellular hand-off.

Explicit loss notification: The major reason of the bad performance of TCP across wireless networks is that it cannot identify the loss of data due to network congestion [43]. This process will help the TCP to determine the loss of data. It also reacts to such types of lost packets based on the present information.

3 Analysis

TCP provides end to end communication and it is also a connection-oriented protocol. After establishing a connection, data can transmit between source and destination. UDP is considered as a simple and connection less protocol and it does not provide end to end communication between source and destination. The data is transmitted in one direction from source to destination without checking the condition of destination. This paper is focused on the comparison of TCP and UDP protocol on the basis of terminologies of data transmission, basic applications and operations.

3.1 Differences in Data Transfer between TCP & UDP

Transmission control protocol offers an ordered and reliable delivery of data between server and users whereas UDP protocol does not provide reliable delivery of data because it is a connection less protocol. TCP and UDP protocols are quite different from each other due to many features that are given below:

3.2 Reliable Delivery of Data

TCP provides a reliable transmission as compared to UDP because it provides acknowledgment and retransmission mechanism in case of packet loss. TCP ensures the reliable delivery of data without the packet loss whereas UDP does not ensure reliable delivery of data.

3.2.1 Ordering of Messages in TCP & UDP

The segments will be transmitted in a sequence in TCP and they are reached at the destination in the same order. If segments will reach in wrong order, TCP has the capability of reordering them. On the other hand, messages are not transferred in sequence during the transmission in case of UDP protocol. It does not provide ordered delivery of the data.

3.2.2 Connection Setup

TCP needs three packets in order to establish a socket connection and handles with the reliability and congestion control. So, it is considered as heavy weight protocol whereas UDP is considered as light weight protocol and not supports ordered delivery of data [1].

3.2.3 Transfer Features

TCP takes data in the form of bit stream and messages are sent through segments whereas UDP sent messages in the form of data grams over the network [3].

3.3 Differences in Basic Operation

3.3.1 On the Basis of Operation, TCP and UDP have Key Differences.

There are three ways to establish a connection such as starting a connection, provide acknowledgments and close the connection [4][7][9]. Once the connection will be made between two devices for communication, data will be sent in both directions [10]. The connection will be closed when transmission is over. When we are comparing the UDP with the TCP, we observe that UDP will not ensure the reliable transmission. Data grams can be duplicated or arrived out of order. It supports broadcasting and multicasting techniques.

3.4 Differences in Applications

There are many applications of TCP such as file transfer and web browsing etc. TCP transmits the segments and controls the segment size and data exchange rate, reads the data in the form of bits provide flow control and prevent the network congestion [8][9]. Congestion control mechanism in TCP connection is also known as window-based transmission which helps to control the network congestion. TCP also supports error correction facilities that lead to a reliable delivery of data [14]. UDP supports time constraint applications. UDP provides in- time delivery of messages but it can compromise on some loss of information. It also supports multicasting and broadcasting. The applications like (TFTP), (DNS) and online games use UDP protocol [20].

4 Conclusion and Future Work

In this paper, we have delved into the intricacies of the transport layer, its protocols, and their fundamental functions. Specifically, we have provided a comprehensive examination of the TCP and UDP

protocols, highlighting their distinctions in terms of data transfer, applications, and operations. Moreover, we briefly explored the operation of the TCP protocol within wireless networks.

Based on our analysis, we conclude that the choice between TCP and UDP depends on specific application requirements. TCP stands out as the preferred protocol when reliability and ordered data delivery are paramount. It excels in scenarios where data integrity and guaranteed transmission are essential. On the other hand, UDP finds its niche in time-sensitive applications, such as real-time streaming and online gaming, where in-time delivery of messages takes precedence over perfect data reliability. UDP's ability to support multicasting and broadcasting makes it suitable for use in applications like internet radio or internet gaming.

Our study has shed light on the unique advantages and limitations of both protocols, demonstrating that they each excel in different scenarios. By offering a detailed comparison of TCP and UDP at the transport layer, we aim to provide a valuable resource for network engineers and application developers when selecting the appropriate protocol for their specific needs.

In closing, our comparative analysis has underscored the importance of selecting the right transport layer protocol based on the specific demands of an application. This research contributes valuable insights into the nuanced differences between TCP and UDP, helping practitioners make informed decisions about protocol selection. As the networking landscape continues to evolve, optimizing TCP and UDP for improved performance remains a promising avenue for future work. Enhancements in these protocols can further enhance their suitability for a broader range of applications, ensuring the continued evolution and efficiency of computer networks.

References

[1] Millán, G., Fuertes, G., Alfaro, M., Carrasco, R., & Vargas, M. (2018, October). A simple and fast algorithm for traffic flow control in high-speed computer networks. In *2018 IEEE International Conference on Automation/XXIII Congress of the Chilean Association of Automatic Control (ICA-ACCA)* (pp. 1-4). IEEE.

[2] Comer, E. D. (2015). *Computer networks and internets: Part 1*.

[3] Kurose, J. F., & Ross, K. W. *Computer Networking*.

[4] Sulistiyarini, D., Ramadhani, D., & Sabirin, F. (2021). Developing Serious Video Games for Data Communication Courses. *JTP-Jurnal Teknologi Pendidikan,* 23(1), 11-22.

[5] Peterson, L. L., & Davie, B. S. (2011). *Computer networks a system approach 5th ed.*

[6] Stevens, W. R., Rudoff, A. M., & Fenner, B. (2003). Unix network programming volume 1: the sockets networking API (Vol. 3). Boston: Addison-Wesley Professional.

[7] Stallings, W. (2007). *Data and computer communications. Pearson Education India.*

[8] Fall, K. R., & Stevens, W. R. (2011). *TCP/IP illustrated, volume 1: The protocols*. addison-Wesley.

[9] Keshav, S., & Kesahv, S. (1997). *An engineering approach to computer networking: ATM networks, the Internet, and the telephone network (Vol. 116).* Reading: Addison-Wesley.

[10] Jacobson, V. (1988). Congestion avoidance and control. *ACM SIGCOMM computer communication review,* 18(4), 314-329.

[11] Sharma, B., & Aseri, T. C. (2012). A comparative analysis of reliable and congestion-aware transport layer protocols for wireless sensor networks. *International Scholarly Research Notices*, 2012.

[12] Keshav, S., & Kesahv, S. (1997). *An engineering approach to computer networking: ATM networks, the Internet, and the telephone network (Vol. 116).* Reading: Addison-Wesley.

[13] Dwyer, C., Hiltz, S., & Passerini, K. (2007). Trust and privacy concern within social networking sites: A comparison of Facebook and MySpace. *AMCIS 2007 proceedings,* 339.

[14] Guerrier, P., & Greiner, A. (2000, January). A generic architecture for on-chip packet-switched interconnections. *In Proceedings of the conference on Design, automation and test in Europe* (pp. 250-256).

[15] Huang, J., Qian, F., Guo, Y., Zhou, Y., Xu, Q., Mao, Z. M., ... & Spatscheck, O. (2013). An in-depth study of LTE: Effect of network protocol and application behavior on performance. *ACM SIGCOMM Computer Communication Review,* 43(4), 363-374.

[16] Martin, R. L. (1995, February). The future of wireless applications. In *Proceedings of 1995 IEEE MTT-S International Topical Symposium on Technologies for Wireless Applications (Conjunction with INTER*

COMM'95) (p. 5). IEEE.

[17] Hiromori, A., Yamaguchi, H., & Higashino, T. (2011, July). A Comprehensive Test Strategy for Network Protocols in Diverse Environment. In 2011 *IEEE 19th Annual International Symposium on Modelling, Analysis, and Simulation of Computer and Telecommunication Systems* (pp. 188-196). IEEE.

[18] Sun, W., Bhaniramka, P., & Jain, R. (2000, November). Quality of service using traffic engineering over MPLS: An analysis. *In Proceedings 25th Annual IEEE Conference on Local Computer Networks.* LCN 2000 (pp. 238-241). IEEE.

[19] Chughtai, H. M. O., Malik, S. A., & Yousaf, M. (2009, December). Performance evaluation of transport layer protocols for video traffic over WiMAX. In *2009 IEEE 13th International Multitopic Conference* (pp. 1-6). IEEE.

[20] Chana, I., & Singh, S. (2014). Quality of service and service level agreements for cloud environments: Issues and challenges. *Cloud Computing: Challenges, Limitations and R&D Solutions*, 51-72.