Special Conference Edition, November, 2018

http://dx.doi.org/10.4314/bajopas.v11i1.69S

Bayero Journal of Pure and Applied Sciences, 11(1): 435 - 442
ISSN 2006 - 6996

MODEL FOR THE ANALYSIS OF IPV4 AND IPV6 PROTOCOLS IN ETHERNET NETWORK

[1]Adamu Aminu and [2]Kurah Anne
[1]Umaru Musa Yar'adua University, Katsina State, Nigeria.
Mathematics and Computer Science Department.
[1]aminu@mail.ru: +2348031581658; [2]anniecute2009@yahoo.com: +2348032846248

ABSTRACT
In this paper, a performance model was presented and some qualities of service (QoS) parameters were used to evaluate the performances of IPv4 and IPv6 protocols in Ethernet network. The QoS parameters used were throughput, end-to-end delay and packet loss. The packet loss was analyzed using iperf as measurement tool; the results of the analysis have shown that IPv4 incurred packet loss of about 12%, while IPv6 incurred almost 0% packet loss. For the analysis of the throughput and en-to-end delay, compacted formulas were presented for their computation and analysis conducted has shown that IPv4 outperforms the IPv6 in terms of throughput and end-to-end delay.

Keywords: IPv4, IPv6, QoS, Throughput, Packet Loss, Delay.

INTRODUCTION

The Internet is a communication platform which is experiencing massive growth and has revolutionized each sector of our society. Such a rapid development of the Internet can be directly linked to the useful services such WWW services, e-mail services, instant messaging, voice over IP, videoconferencing etc. it provides to users. It is expected to have many more services in the future since Internet uses open protocols that are freely available to services developers.

The IPv4 protocol was one of the key protocols defined at the network layer of the Internet model and until recently was the major protocol used for host-to-host communication between end systems over the Internet (James and Keith, 2010; Forouzan, 2007; Hinden, 1998; Ali, 2012). One of the shortcomings of the IPv4 protocol is the limited address space it provides. The IPv4 uses 32-bit address to uniquely and universally identify each device connected to the Internet, however, the 32-bit address provided by IPv4 protocol means that the total address space provided is 2^{32}, approximately 4.3 billion addresses, on inception this number of addresses was seemed to be adequate for the then and future devices, but the time proved that assumption wrong. Considering the array of devices such as PC, laptops, sensors, tablets, smartphones, web cameras etc., which could potentially be connected to the Internet, the IPv4 address system is very much unsuitable for the fast growing Internet. Despite the short term solutions created such as NAT, sub netting and classless addressing to boost the IPv4 address space, address depletion is still unavoidable considering the trend of the Internet growth. Additionally, nowadays the Internet is widely used for the provision of real time audio and video services, such services are delay intolerable, however, the IPv4 protocol does not provide clever strategies to minimize delay and to reserve resources for class of traffic with some priorities. In addition to the above IPv4 shortcomings, nowadays security is of utmost important, unfortunately no encryption or authentication provided by IPv4 protocol.

To address these shortcomings of IPv4 protocol, Internetworking protocol next generation (IPng), also known Internetworking protocol version 6 (IPv6) was designed to accommodate the unforeseen development of the Internet (James and Keith, 2010; Forouzan, 2007; Hinden, 1998; Ali, 2012). The large address space provided, being 128-bit address protocol, an increase of 2^{96} in the address space over that of IPv4 is a huge increment. IPv6 provides better header format for the simplification and speeding of routing process, provides supports for resource allocation for traffics with different priorities and provides better security by providing confidentiality and integrity of data transmitted over a network. All these are some of the advantages of IPv6 over the IPv4 protocol (James and Keith, 2010; Forouzan, 2007; Hinden, 1998; Ali, 2012).

IPv6 is being massively adopted which may soon lead to the extinction of IPv4 protocol (Wang, et al., 2005).

However, it is paramount to analyze the performance of these protocols in an Ethernet network in terms of some basic network QoS parameters such as throughput, delay and packet loss (Gozdecki, et al., 2003; Wang, et al., 2005; Eric and Neudith, 2011). In this paper a tractable model for the analysis of IPv6 and IPv4 protocols over Ethernet network was presented and analysis was conducted via analytical and measurement techniques. An analytical model for such purpose was first presented by (Eric and Neudith, 2011), however, in this paper similar model was studied and compacted formulas were presented for the computation of QoS parameters, additionally, measurement experiment was conducted for the analysis of packet loss, iperf (iperf, 2018) was used as a measurement tool. In section II, a tractable model was presented for the analysis of end-to-end delay and throughput for the Ethernet network, compacted formulas were presented to ease the analysis. Section III presents the measurement and analysis results and section IV concludes the paper.

I. Model for the Analysis of End-to-End Delay

In this section the end-to-end delay would be analyzed for both the IPv4 and IPv6 protocols. The end-to-end delay consists of three components i.e. transmission, processing and propagation delays (James and Keith, 2012; Forouzan, 2007). Now let's denote by $\Delta_x(n)$, $x \in \{\mathrm{IPv4}, \mathrm{IPv6}\}$ the end-to-end delay for the transmission of *n*-bytes of IP payload from d_0 to d_m device connected through the chain of routers $d_i, i = 1, \ldots, m-1$ (Figure 1).

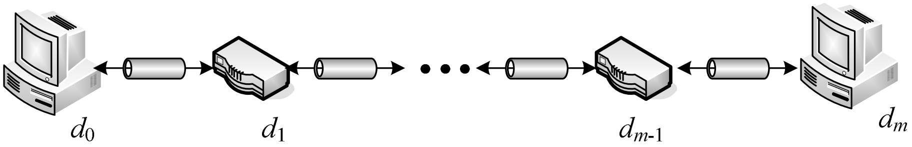

Figure 1. Network connections

To compute $\Delta_x(n)$, $x \in \{\mathrm{IPv4}, \mathrm{IPv6}\}$, let's denote by $\upsilon_x(n \mid B)$, $x \in \{\mathrm{IPv4}, \mathrm{IPv6}\}$ the transmission delay for *n* bytes of IP payload over the Ethernet network with bandwidth $B > 0$. Further, let's denote by $\varepsilon(d_i, d_j)$, $i, j = 0, \ldots, m$, the propagation delay between d_i and d_j, $i \neq j$ and $j - i = 1$. Denote by $\tau(d_i)$, $i = 0, \ldots, m$, the processing delay at d_i, $i = 0, \ldots, m$. Therefore the generalized formula of end-to-end delay for transmission of *n* bytes of IP payload between d_0 and d_m devices connected through the chain of routers $d_i, i = 1, \ldots, m-1$ is given by

$$\Delta_x(n) = \sum_{i=0}^{m} \tau(d_i) + \sum_{i=0}^{m} \upsilon_x(n \mid B_i) + \sum_{i=0}^{m}\sum_{j=0}^{m} 1(i,j)\,\varepsilon(d_i, d_j) \quad (1)$$

where $1(i,j) = \begin{cases} 1 & \text{if} \quad j - i = 1 \\ 0 & \text{otherwise} \end{cases}$, $x \in \{\mathrm{IPv4}, \mathrm{IPv6}\}$.

To further analyze the end-to-end delay, let's assume that the network consists of homogeneous under-loaded devices (routers of the same Ethernet technology with $B_i = B, \forall i = 1, \ldots, m-1$) connected together with equal connection length. These assumptions implied that the propagation delay is the same across all the links ($\varepsilon(d_i, d_j) = \delta, j - i = 1$) and the processing delay across all the devices is negligible, i.e. $\tau(d_i) = 0$, $i = 0, \ldots, m$.

Hence, (1) is transformed to

$$\Delta_x(n) = (m-1) \cdot (\upsilon_x(n \mid B) + \delta), \quad (2)$$

where $x \in \{\mathrm{IPv4}, \mathrm{IPv6}\}$.

Let's obtain the expression for transmission delay for IPv4 and IPv6 in form of proposition 1 and proposition 1 respectively.

Proposition 1. The minimum time required to transmit *n* bytes of IPv4 datagram over the considered Ethernet network with bandwidth $B > 0$ is

$$\upsilon_{\mathrm{IPv4}}(n \mid B) = \begin{cases} \dfrac{576}{B} & \text{if } n < 26 \\ \dfrac{(n+46) \cdot 8}{B} & \text{if } n \geq 26 \end{cases} \quad (3)$$

■ **Proof.** Let's consider the IPv4 datagram format. The IPv4 datagram is of variable packet length and has two (2) parts: header part and data or payload part. The header length range from 20 to 60 bytes and it contains necessary information for routing and effective delivery. The header consists of two parts: the fixed part and the variable part. The fixed part of the header is 20 bytes long (Figure 2); however, the variable part comprises the options which can be a maximum of 40 bytes. Hence, the size of IPv4 header extends from 20 bytes to the maximum of 60 bytes when options are considered (James and Keith, 2012; Forouzan, 2007; Hinden, 1998). However, header options were meant to be used rarely - hence in most cases the options is not used in datagram headers to save overhead. Finally, assume no option is used, then the IPv4 datagram header is chosen to be 20 bytes throughout the paper for IPv4 protocol.

The IPv4 datagram header in 4-byte section format is shown in Figure 2. The fields in the IPv4 datagram header are shown in Table 1.

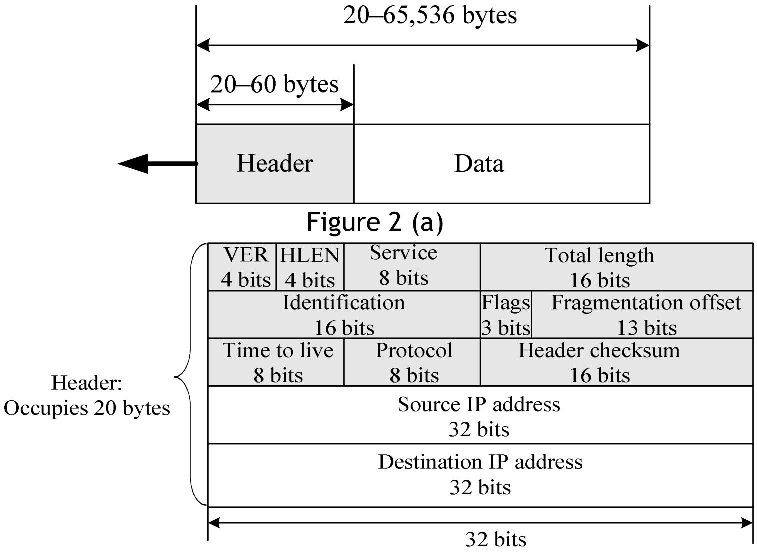

Figure 2 (a)

Figure 2 (b)

Figure 2. IPv4 datagram format

Table 1. IPv4 datagram fields (James and Keith, 2012; Forouzan, 2007; Hinden, 1998).

Field	Size (Bits)	Description
Version (VER)	4	Defines the version of IP protocol
Header Length	4	Defines the total length of the datagram header
Type of Service	8	To allow different types of IP datagrams (for example, datagrams particularly requiring low delay, high throughput, or reliability) to be distinguished from each other
Datagram Length	16	This is the total length of the IP datagram (header plus data), measured in bytes. Since this field is 16 bits long, the theoretical maximum size of the IP datagram is 65,535 bytes. *However, datagrams are rarely larger than 1,500 bytes*
Identifier	16	Used in fragmentation
Flags	3	Used in fragmentation
Fragmentation Offset	13	Used in fragmentation
Time-to-live	8	The time for the datagram to stay in the network
Protocol	8	It indicates the specific transport-layer protocol to which the data portion of the IP datagram should be passed when it reached the final destination
Header Checksum	16	It aids a router in detecting bit errors in a received IP datagram
Source Address	32	Source address of the IP datagram
Destination Address	32	Destination address of the IP datagram
Total	160 bits (20 bytes)	

By summing the length of each field in the IPv4 datagram header, it can be seen that the header length is 20 bytes (no option considered).

Since we are considering Ethernet network, further let's consider the format of an Ethernet frame shown on Figure 3.

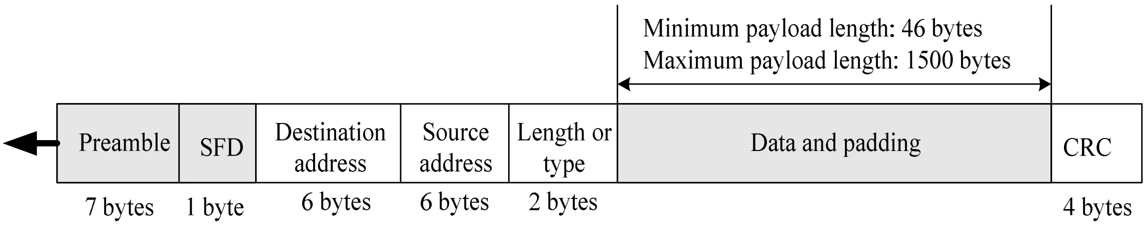

Figure 3. Ethernet frame

The detail description of each field contained in the Ethernet frame is provided in table 2.

Table 2. The detail description of each field contained in the Ethernet frame (James and Keith, 2012; Forouzan, 2007; Hinden, 1998).

Field	**Size (Bytes)**	**Description**
Preamble	7	It is an alternating 0s and 1s which alerts the receiving device to the coming frame
Start Frame Delimiter (SFD)	1	Signals the beginning of a frame (10101011)
Destination address	6	Contains the physical address of the device to receive the frame
Source address	6	Contains the physical address of the device which send the frame
Length or Type	2	Used to define the upper-layer protocol using MAC frame
CRC	4	Contains error detection information
Total	*26 bytes*	
Data (Datagram size encapsulated from the upper layer protocol)	Min = 46 and Max = 1500 (including the 20 bytes datagram header)	
Total frame size	If Min is considered is (46 bytes + 26 bytes = 72 bytes) If Max is considered is (1500 bytes + 26 bytes = 1526 bytes)	

The total of the minimum data length from the upper layer (network layer) is
72 bytes – 26 bytes = 46 bytes,
If the upper layer data is less than 46 bytes, padding is added to make up the differences. Considering the fact that the header length of IPv4 is 20 bytes (without option), therefore, the minimum length of the actual data is
46 bytes – 20 bytes = 26 bytes,
Let's assume that the data length from the upper layer is $n < 26$, $n+20 < 46$, hence padding is required (James and Keith, 2012; Forouzan, 2007; Eric and Neudith, 2011). The length of data to be used for the padding is $26-n$ bytes, in order to make up to the minimum required data length, i.e. 46 bytes (Figure 3).
Hence, $26+(20+n+26-n)=72$ bytes.
When $n<26$, the total data length transmitted is $72\times 8 = 576$ bits. Then the time required for transmission is

$\upsilon_{IPv4}\left(n\,|\,B\right)=\dfrac{576}{B}$, when $n<26$ and $B>0$.

For $n \geq 26$ bytes, no padding is needed and the maximum data length from the upper layer for transmission is 1,480 bytes without the header length.
Hence, the total data length to be transmitted through the network is $\left(46+n\right)\cdot 8$ bits. Then the time required for transmission is

$\upsilon_{IPv4}\left(n\,|\,B\right)=\dfrac{\left(46+n\right)\cdot 8}{B}$, when $n \geq 26$ and $B>0$.■

Proposition 2. The minimum time required to transmit *n* bytes of IPv6 datagram over the considered Ethernet network with bandwidth $B>0$ is

$$\upsilon_{IPv6}\left(n\,|\,B\right)=\begin{cases}\dfrac{576}{B} & \text{if } n<6\\[2ex] \dfrac{(n+66)\cdot 8}{B} & \text{if } n\geq 6\end{cases} \qquad (4)$$

■ **Proof.** Let's consider the IPv6 datagram format. It consists of mandatory base header and the payload, the payload however, is divided into two parts optional extension field and the data from the upper layer. The base header length is 40 bytes and the extension header plus the data from the upper layer is up to 65535 bytes (Figure 4).

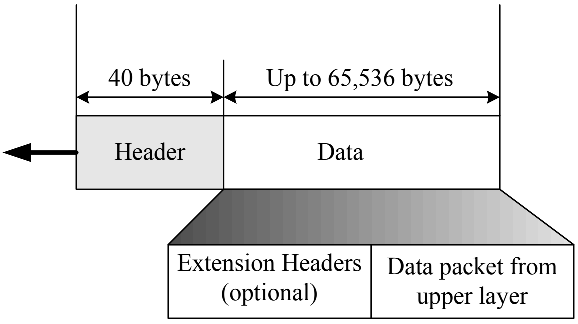

Figure 4 (a)

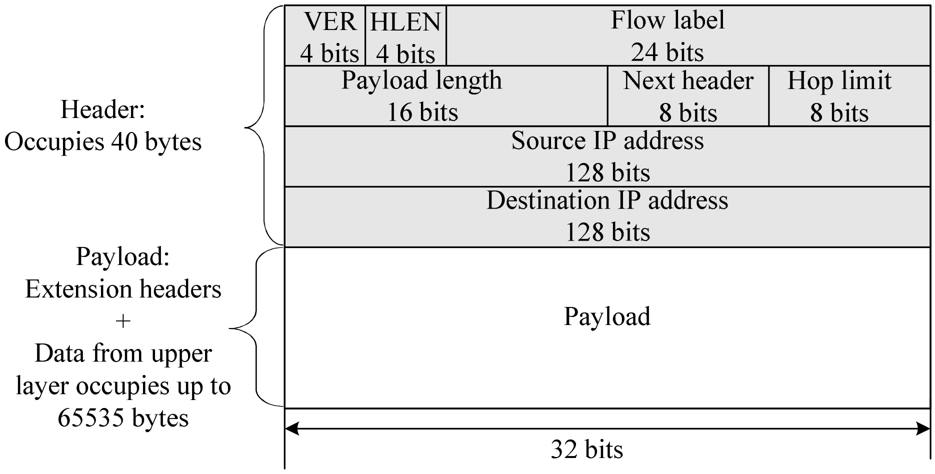
VER 4 bits
HLEN 4 bits
Flow label 24 bits
Payload length 16 bits
Next header 8 bits
Hop limit 8 bits
Source IP address 128 bits
Destination IP address 128 bits
Header: Occupies 40 bytes
Payload: Extension headers + Data from upper layer occupies up to 65535 bytes
Payload
32 bits

Figure 4 (b)

Figure 4. IPv6 datagram format

Further, let's consider the eight (8) fields contained in the base header, so as to get the total length.

Table 2. IPv6 datagram fields (James and Keith, 2012; Forouzan, 2007; Hinden, 1998).

	Field	Size (Bits)	Description
	Version (VER)	4	Defines the version of IP protocol, in this case the value is 6
	Priority	4	Defines the priority of the packet in case of congestion
	Flow label	24	Created to provide special handling for a particular flow of data
	Payload length	16	Defines the length of the IP datagram without the length of the base header
	Next header	8	Defines the header that follows the base header in the datagram (same as protocol field in IPv4 case)
	Hop limit	8	Identifies the time to live for the datagram (same as TTL field in IPv4 case)
	Source address	128	Source address of the IP datagram
	Destination address	128	Destination address of the IP datagram
Total		320 bits (40 bytes)	

The total of the minimum data length from the upper layer (network layer) is
72 bytes – 26 bytes = 46 bytes
If the upper layer data is less than 46 bytes, padding is added to make up the differences. Considering the fact that the header length of IPv6 is 40 bytes (without option), therefore, the minimum length of the actual data is 46 bytes – 40 bytes = 6 bytes.
Let's assume that the data size from the upper layer is $n < 6$ bytes, $n + 40 < 46$ bytes, hence padding is required (James and Keith, 2012; Forouzan, 2007; Eric and Neudith, 2011). The size of data to be used for the padding is $6 - n$ bytes, in order to make up to the minimum required data length, i.e. 46 bytes (Figure 3).
Hence, $26 + (40 + n + 6 - n) = 72$ bytes. When $n < 6$, the total data transmitted is $72 \times 8 = 576$ bits. Then the time required for the transmission of $n < 6$ bytes of IPv6 datagram is

$\upsilon_{IPv6}(n \mid B) = \frac{576}{B}$, when $n < 26$ and $B > 0$.

For $n \geq 6$ bytes, no padding is needed and the maximum data length from the upper layer for transmission is 1,460 bytes without the header length (Figure 3).
Hence, the total data length to be transmitted through the network is $(66 + n) \cdot 8$ bits. Then the time required for transmission is

$\upsilon_{IPv6}(n \mid B) = \frac{(66+n) \cdot 8}{B}$, when $n \geq 26$ and $B > 0$.■

Proposition 3. The maximum achievable throughput in transmitting *n* bytes of IP payload over the considered Ethernet network is

i. For homogeneous network

$$\eta = \frac{n}{\upsilon_x(n \mid B)} \cdot 8, \tag{5}$$

ii. For heterogeneous network, i.e. network between the two corresponding devices consists of Standard Ethernet, Fast Ethernet or Gigabit Ethernet. For this case let $B_{\min} = \min\{B_i\}_{i=1,\mathrm{L}\ ,m-1}$. Hence

$$\eta = \frac{n}{\upsilon_x(n \mid B_{\min})} \cdot 8, \tag{6}$$

where $x \in \{\mathrm{IPv4}, \mathrm{IPv6}\}$, $T_x(n \mid B_{\min}) \neq 0$.

II. Experiments and analysis

Performance analysis was firstly conducted via measurement technique. In this case, for the purpose of the experiment (Iperf3, 2018) was used as a software measurement tool, which was installed in two identical PCs connected by point-to-point link using a cross over cable, the 10Base-T Ethernet was used as the LAN technology. Both the PCs support the IPv4 and IPv6 protocols. The Iperf's statistics were produced at the server instance of the Iperf traffic generator; however, the TCP protocol was by used as a transmission protocol. The results for the analysis of packet loss have shown that IPv6 protocol has 0% packet loss, where IPv4 introduces 16% packet loss.
Further let's analyze the other QoS parameters i.e. end-to-end delay and throughput using the analytical model, the 10Base-T Ethernet was used and δ = 30µs. Firstly, the graph in Figure 5 has shown that the throughput of IPv4 protocol is higher than that of the IPv6 protocol, however, it can also be observed that as the data size increases the throughput also increases.

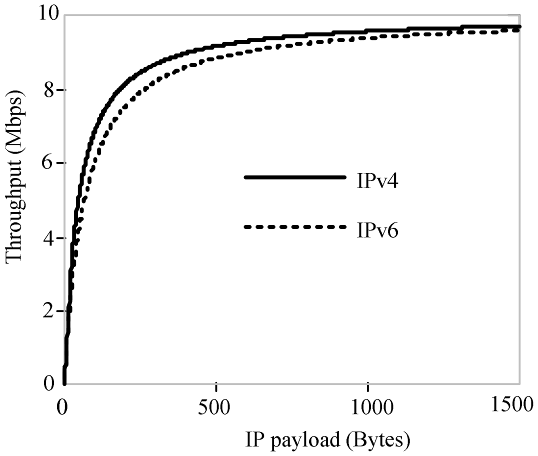

Figure 5. Throughput for IPv4 and IPv6

The graphs in Figure 6 shows the end-to-end delay for IPv4 protocol (Δ_{IPv4}) for different values of *m*, $m = 1, \ldots 6$. From the graphs, it can be observed that for all the value of *m*, as the size of payload transmitted increases, Δ_{IPv4} increases linearly. However, results of the analysis revealed that IPv6 protocol introduces more delay than IPv4 protocol, i.e. $\Delta_{\text{IPv6}} > \Delta_{\text{IPv4}}$.

$$\Delta_{\text{IPv6}} = m \cdot 2^4 + \Delta_{\text{IPv4}} \qquad (7)$$

Denote by $D(n,m) = \Delta_{\text{IPv6}} - \Delta_{\text{IPv4}}$, however, results of the analysis have shown that $D(n,m) = D(n_\bullet, m)$, $\forall m$ and $n_\bullet \neq n$, hence, $D(n,m) = D(m)$, from graph in Figure 7, it can be seen that as *m* increases, $D(m)$ also increases. Then for the considered network, $D(m)$ will have the following form:

$$D(m) = m \cdot 2^4 \qquad (8)$$

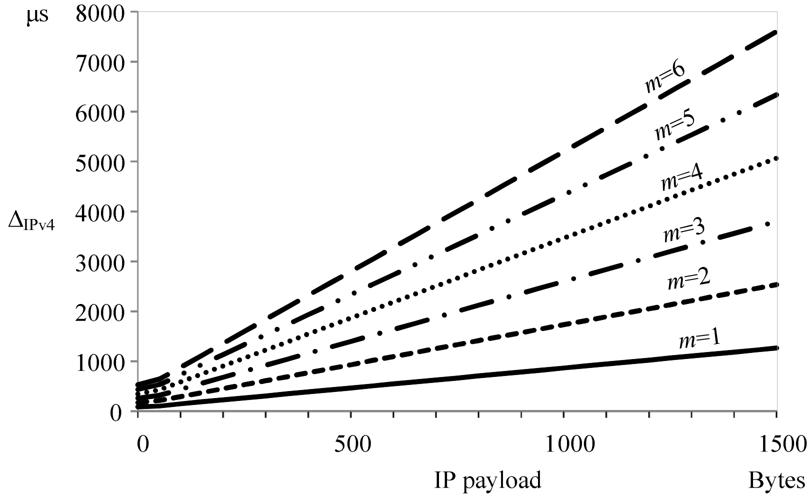

Figure 6. Delay for IPv4 protocol

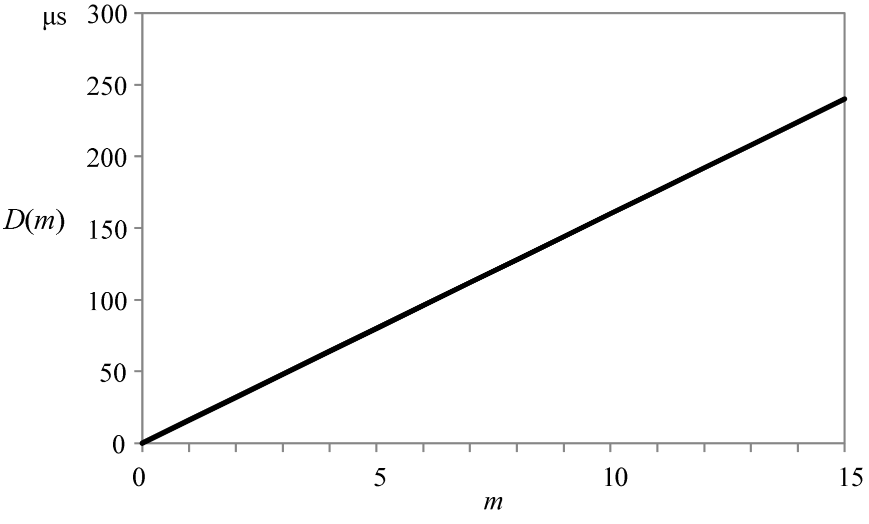

Figure 7. IPv4 and IPv6 delay differences

Conclusion

In this paper analysis of IPv6 and IPv4 protocols was conducted for 10Based-T Ethernet network, during the analysis some key network QoS parameters (i.e. packet loss, throughput and end-to-end delay) were considered as performance measures. Compacted expressions were formulated for the computation of the two considered QoS parameters. For the analysis of packet loss, measurement experiment was conducted using iperf as a measurement tool. The results of the analysis have shown that IPv4 incur more packet loss than the newly invented IPv6 protocol, on the other hand, the analysis revealed that IPv4 outperform IPv6 in terms of throughput and end-to-end delay.

References

James, E. K. & Keith, W. R (2012). Computer networking: a top-down approach featuring the Internet. Book, 6th Edition.

Forouzan, A. B. (2007). Data Communication and Networking. Book, 4th Edition.

Hinden, R. and Deering, S. (1998). Internet Protocol, Version 6 (IPv6) Specification. RFC 2460, 1998, Online: https://www.tools.ietf.org/html/rfc2460.

Ali N. A. (2012). Comparison Study between IPv4 & IPv6. International Journal of Computer Science Issues, 9, 3, 314-317.

Gozdecki J., et al. (2003). Quality of Service Terminology in IP Networks. Communications Magazine, IEEE, 41, 153-159.

Wang, Y., et al. (2005). Understanding current IPv6 performance: A measurement study. Proceedings of the 10th IEEE Symposium on Computers and Communication Washington: IEEE Computer Society, 71-76.

Eric, G., and Neudith, M. (2011). Modeling IPv4 and IPv6 Performance in Ethernet Networks. International Journal of Computer and Electrical Engineering, 3, 283-288.

Iperf (2018). Homepage, http://dast.nlanr.net/Projects/Iperf/.