Estimation of Perceived Quality of Service for Applications on IPv6 Networks

Xiaoming Zhou
Delft University of Technology
P.O. Box 5031, 2600 GA Delft
X.Zhou@ewi.tudelft.nl

Rob E. Kooij
TNO ICT
P.O. Box 5050, 2600 GB Delft
R.E.Kooij@telecom.tno.nl

Henk Uijterwaal
RIPE NCC TTM
Singel 258, 1016 AB, Amsterdam
Henk@ripe.net

Piet Van Mieghem
Delft University of Technology
P.O. Box 5031, 2600 GA Delft
P.VanMieghem@ewi.tudelft.nl

ABSTRACT

To provide high quality service to future Internet applications, IPv6 performance measurements are needed. However, to the best of our knowledge, IPv6 delay and loss performance evolution and their impact on applications have not been studied on a large scale. In this paper, we have analyzed more than 600 end-to-end IPv6 paths between about 26 testboxes of RIPE NCC over the past two years, and compared the delay and loss performance evolution in IPv6 with their IPv4 counterparts. We present and discuss the measurement methodology, and we provide evidence that IPv6 network has a higher delay and loss evolution than IPv4. Finally, based upon our measurements, we assess the perceived quality of three real-life applications: VoIP, Video-over-IP and data communication services based upon TCP.

Categories and Subject Descriptors

D.2.5 [**Computer-Communication Networks**]: Local and Wide-area Networks – *Internet (e.g., TCP/IP).*

General Terms: Measurement

Keywords: QoS, Measurement, IPv6

1. INTRODUCTION

Currently, IPv6 (RFC 2460) is moving continuously towards commercial deployment. To provide high quality service to future Internet applications, insight in IPv6 performance measurements is needed. However, to the best of our knowledge, IPv6 delay and loss performance evolution and their impact on applications have previously not been studied on a large scale. Earlier studies mainly focus on IPv6 transition technologies [1] or identifying IPv6 network problems in the dual-stack world by measurements from a few days [2]. Because compared to IPv4, IPv6 is still in its infancy and it is hardly ever used by real-life applications, there is a lack of knowledge about the network performance of end-to-end IPv6 communication. Studying the large-scale IPv6 delay and loss performance evolution and their impact on applications are important for ISPs to understand the performance of the current IPv6 networks, and to provide high quality service to future Internet applications.

In this paper, we investigate the large-scale IPv6 network performance evolution in terms of packet delay and loss over the last two years and their impact on real-life applications.

The data set for our study was obtained through measurements conducted in the RIPE NCC TTM[1] project. At the time of writing, this data set contains active measurements between a set of about 26 test boxes supporting IPv6.

Our contribution is threefold. First, we present a measurement methodology to evaluate the IPv6 evolution performance by comparing IPv6 and IPv4 performance on a path-by-path basis (Section 3). Second, we provide evidence that IPv6 has a large delay and loss compared to the IPv4 counterparts (Section 4). Third, we estimate and discuss the users' perceived quality of applications based on the measurements, and give evidence that IPv6 core networks may achieve a good quality for the real-life applications Voice-over-IP, Video-over-IP and data communication services based upon TCP (Section 5).

2. THE TRANSITI ON TECHNIQUES FROM IPV4 TO IPV6

IPv6 allocates 128 bits to represent an address, while IPv4 only allocates 32 bits. The number of different combinations therefore increases from 2^{32} to 2^{64} networks of 2^{64} addresses, which reduces the need for network address translation (NAT). The use of NAT sustains the explosion of end devices, but also greatly increases network complexity, which is a big barrier to the widespread introduction of point-to-point applications. In addition, IPv6 also offers other advanced capabilities with respect to security, autoconfiguration and mobility.

Since a world-wide scale migration from IPv4 to IPv6 within a short period is unfeasible, three main transition techniques were developed to make the continuous transition from the current IPv4 Internet to IPv6 possible.

The first technique is the dual-stack network. It requires hosts and routers to implement both the IPv4 and IPv6 protocol. This enables networks to support both IPv4 and IPv6 services and applications during the transition period. At the present time, the

PM²HW²N'06, October 6, 2006, Torremolinos, Malaga, Spain.

[1] www.ripe.net/ttm

dual-stack approach is possible to achieve a relatively good performance [1].

The second technique relies on tunnelling. Tunnelling enables new IPv6 networking functions while still preserving the underlying IPv4 network as is. For instance, when an IPv6 packet is leaving an IPv6 domain and entering an IPv4 domain, the packet is encapsulated in an IPv4 packet by a border router and transmitted through the network. When the packet reaches the other end of the IPv4 network, it is decapsulated at the border of the receiving IPv6 network. Tunnels can be statically or dynamically configured. 6to4 (RFC 3056) is a technique to transport IPv6 traffic over IPv4 networks without the need for automatic or configured tunneling. 6over4 (RFC 2529) is another technique that uses an existing IPv4 domain with multicast support to create a virtual link-layer for IPv6 hosts. Tunnels over Generic Routing Encapsulation (GRE) (RFC 2473 and RFC 1701) have an extra encapsulation header to enable IPv6 traffic forwarding over an existing IPv4 infrastructure, with minimum changes.

The last technique uses a proxy and translation mechanism. Translation is necessary in case no other methods like tunneling nor native IPv6 is available, and an IPv6-only host still wants to communicate with an IPv4-only host. An example of such a technique is NAT-PT (RFC 2766), which performs address and protocol translation at the borders between non-homogeneous networks at the IP level.

With different IPv6 transport mechanisms, IPv6 connectivity across the backbone can be set up with multiple segments managed independently. For example, an enterprise could decide to deploy a dual-stack network to connect to an ISP with a native IPv6, while connecting to another ISP with IPv6 over IPv4 tunnels. Since different organizations are at different stages in their transition to IPv6, we have a mix of native paths and tunnels as well as a mix of single and dual-stack nodes today.

3. MEASUREMENT METHODOLOGY

3.1 Definitions of One-way Delay and Loss

The motivation of understanding one-way packet delay and loss from a source (Src) to a destination (Dst) is driven by some reasons: like excessive packet delay or loss (relative to some threshold value) could degrade perceived quality of certain real-time applications. The larger the value of packet delay and loss, the more difficult it is for transport-layer protocols to sustain high throughput.

A packet has a one-way delay ΔT ($\Delta T > 0$) from Src to Dst if Src sent the first bit of the packet at time T and Dst received the last bit of the packet at time $T + \Delta T$. The minimum delay reflects propagation and transmission delay, and also reflects the delay that will likely be experienced in the slightly loaded path. If a packet fails to arrive within a reasonable period of time (such as 10 seconds), the one-way delay is taken to be undefined (RFC 2679). The delay between two nodes is the result of many factors, for example the geographical distance, the number of hops, the load and capacity of the links, the policy routing decisions made on the path and even the way IP packets are transported in the layer 2 architecture.

The one-way packet loss from Src to Dst at T is defined as 0 if Src sent the first bit of a packet at time T and Dst received that packet. The one-way packet-loss is exactly zero when the one-way delay is a finite value, and 1 when the one-way delay is undefined. Packet loss occurs where network traffic fails to reach its destination within a reasonable period of time. It may be due to congestion of the network, or the change between a source-destination path.

3.2 Experimental Setup: RIPE NCC TTM

Our data is provided by the RIPE TTM project. At the time of writing, the TTM infrastructure (which is solely in the core network) consists of approximately 26 IPv6 measurement boxes scattered over Europe, Asia and USA. As RIPE NCC is connected to the Amsterdam Exchange Point, it maintains many IPv6 peers with other 6net participants, using BGP4+ as Exterior Gateway Protocol (EGP). Between each path of measurement boxes, both IPv6 and IPv4 UDP packets of a fixed payload (100 bytes), called probepackets, are continuously transmitted with interarrival times of about 30 seconds, resulting in a total of about 2886 robepackets between each path per day. The sending measurement box generates an accurate time-stamp synchronized via GPS in each probe-packet, while the receiving measurement box reads the GPS-time of the arrival of the probe packet.

The end-to-end delay is defined as the difference between these two time-stamps and has an accuracy of about 10 μs. The hopcount of a path between two measurement boxes is measured every 6 minutes using traceroute. For simplicity, we assumed that those packets between two traceroute measurements did not change the traceroute path.

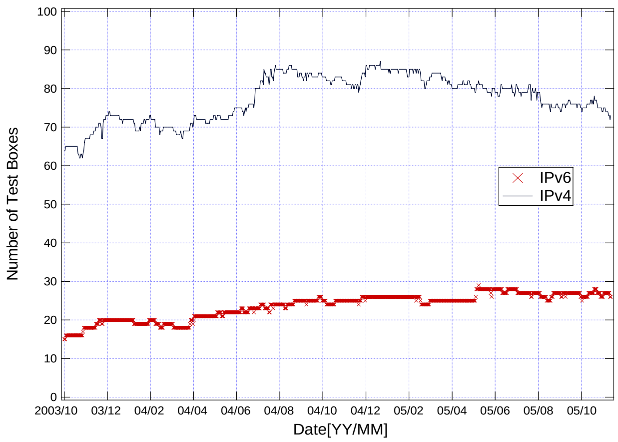
Furthermore, a Maximum Transmission Unit (MTU) detection algorithm is run once per hour. In addition to the traceroute measurements, we use a tunnel discovery tool to identify different tunnels on those IPv6 paths. The tunnel discovery tool detects an IPv6 tunnel by measuring the MTU size (normally 1500 bytes) over an entire path. If a path contains a tunnel, the MTU on that path will usually be lower than 1500, since extra headers are added to the packet. However, this method is not perfect as not all links have an MTU 1500. Another problem is that the tunnel discovery tool cannot detect more than one tunnel. For instance, if there is an IPv6-in-IPv4 tunnel (MTU 1480) followed by a GRE tunnel (MTU 1476 and 1472) or BSD tunnels (MTU 1280), the tool can only detect the tunnel with the lowest MTU.

3.3 Research Challenge

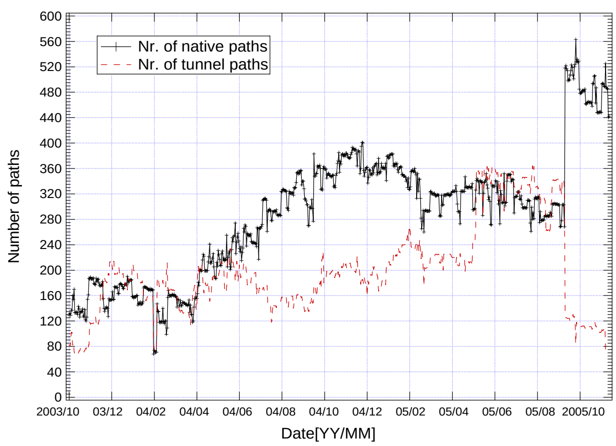
Before presenting the analysis, we formulate the two research challenges.

The first difficult research challenge lies in analyzing the big measurement database. We have analyzed more than 400 GB zipped data collected over 2 years (from 1 Oct. 2003 to 31 Oct. 2005). Fig. 1.(a) shows the numbers of active IPv6 and IPv4 testboxes in the TTM infrastructure over 2 years. Note that not all boxes are active all the time due to reasons such as system updates or failures. The number of IPv4 paths is much larger than that of IPv6 counterparts. Fig. 1.(a) also shows that the numbers of active IPv6 testboxes and paths have been steadily increasing over time. On Oct. 1, 2003, there were 15 active IPv6 testboxes with 210 active IPv6 source-destination paths; by the end of Oct. 2005, these numbers have increased to 29 and 811, respectively. For a fair comparison, only those testboxes supporting both IPv4 and IPv6 traffic were selected in our study.

The second challenge is that the high dynamic evolution of IPv6 tunnels adds analysis complexity. Some IPv6 tunnel paths changed to native paths, and vice versa. Fig. 1.(b) shows the numbers of active native and tunnel paths over the last 2 years. It also shows that in Oct. 2003 about 61% of the total IPv6 paths were native paths, while by the end of Oct. 2005, this number increased to 86%. We observe that there were 328 IPv6 tunnel paths before 30 Aug. 2005, while about 31% of those paths have changed to IPv6 native paths after that, since several ISPs upgraded their routers to dual-stack for better performance.



(a) nr. of active testboxes over time



(b) nr. of native and tunnels over time

Figure 1. The dynamic evolution of IPv6 paths.

Although IPv6 will replace IPv4 in the future, it is expected that IPv4 and IPv6 hosts will coexist for a substantial time during the steady migration from IPv4 to IPv6. It is important to understand how to measure and test IPv6 native/tunneling performance. To qualify the current IPv6 infrastructure, we use IPv4 performance as a comparison base, and compare the IPv6 delay and loss performance and the corresponding IPv4 performance in a path-to-path basis.

4. DELAY AND LOSS PERFORMANCE

4.1 Evolution of Delay Performance of all TTM paths over two years

Since minimal, average, and maximal IPv6 and IPv4 delay are sensitive to the clock error, to minimize this error, for each Src-Dst path, 2.5 percentile, median and 97.5 percentile delay are shown instead. For each Src-Dst path *i*, we first made the delay histogram distribution over a time interval (e.g. one day). Then, we computed the 2.5 percentile $D_{2.5}(i)$, median $D_{50}(i)$ and 97.5 percentile $D_{97.5}(i)$ delay values for that path. We repeated this experiment for all the paths. When all paths were computed, we presented the average values of the 2.5 percentile, median and 97.5 percentile values of all paths. For example, $\frac{1}{n}\sum_{i=1}^{n} D_{2.5}(i)$ is shown in the following graphs of 2.5 percentile values, where *n* is the number of paths in that time interval.

Fig. 2.(a), (b) and (c) show the average 2.5 percentile, median and 97.5 percentile over two years with one day intervals, and Fig. 2.(d) shows the loss comparison between IPv6 and IPv4. Fig. 2.(a), (b) and (c) show that on average IPv6 has about 61% higher 2.5 percentile delay than the IPv4 counterparts, while about 64% and 157% for the cases of median and 97.5 percentile delays, respectively. Hence, both average and variation of the delay are worse for IPv6. Fig. 2.(a)(b) show that since late 2003, IPv6 has a larger average delay than the IPv4 counterpart. Over time, the latter has been steadily and slightly decreasing, however, the former shows a relative big variation, but not much affected over the time.

Fig. 2.(d) shows that the packet loss of all paths over the two year between IPv4 and IPv6 are small (less than 0.02%), and do not change much over time. The main reason lies in the experimental setup: we send the probe packets (100 bytes UDP packets) about every 30 seconds in the TTM infrastructure, which is solely in the core network. On the other side, our results show that over the years, the IPv6 loss is about 1 order of magnitude larger than the IPv4 loss.

Based on the above measurements, we conclude that even though new testboxes were added into the measurement testbed in two years, the IPv6 and IPv4 delay and loss did not change significantly, and IPv4 outperformed IPv6 in term of delay and loss during the whole period.

In an accompanying paper [10] we give evidence that IPv6 tunnels degrade the network delay performance and could explain the reason why IPv6 is outperformed by IPv4.

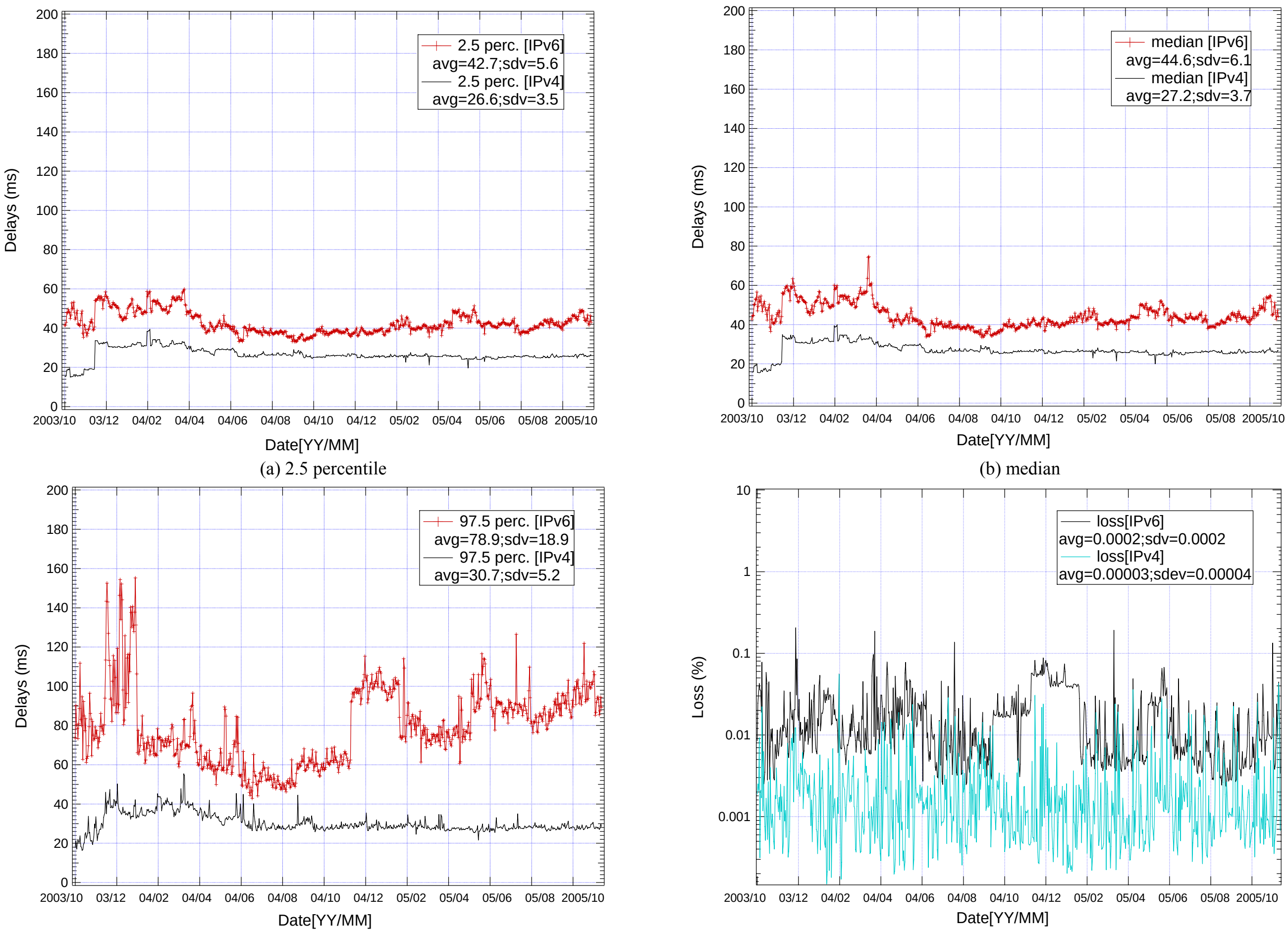
5. ESTIMATION OF PERCEIVED QUALITY OF SERVICE

In the previous section we have compared the network performance of IPv4 and IPv6 as it evolved over a period of two years. However to end-users, it is not the network performance that matters most to them but the perception of the quality of applications running over the network. In general it can be stated that the large scale deployment of applications will only be successful if the perceived quality of these applications will be sufficiently high.

In this section we will investigate how the differences in network performance for IPv4 and IPv6 translate to the perceived quality domain. In particular we will assess the perceived quality of three real-life applications: Voice-over-IP, Video-over-IP and data communication services based upon TCP.

5.1 Assessment of perceived quality of VoIP

The E-Model (ITU-T Recomm. G.107) was used to estimate the perceived quality of voice. Every rating R-value calculated



(a) 2.5 percentile

(b) median

(c) 97.5 percentile

(d) packets loss

Figure 2. Performance of IPv6 paths and the IPv4 counterparts over time.

from the E-Model corresponds to a Mean Opinion Score (MOS) value (as shown in Table 1), and it is used to predict subjective user reactions. An R-value above 70 corresponds to PSTN quality.

Table 1. Speech transmission quality classes and corresponding R-value rang

R-value	100>R>90	90>R>80	60>R>70	R<60
MOS	4.5-4.34	4.34-4.03	4.03-3.60	<3.60
Quality	best	high	medium	low

The mapping function from an R-value to a MOS value has the following form [3]:

$$MOS \quad = \quad 1 + 0.035R + 7\times 10^{-6} R(R-60)(100-R)$$

where the output of the E-Model is a transmission rating factor R:

$$R \quad = \quad (R_0 - I_s) - I_d - I_e + A$$

where R_0 is the effect of background and circuit noise, while I_s captures the effect of quantization. Both R_0 and I_s describe the transmitted voice signal itself and do not depend on the transport network. I_d is the impairment caused by one-way delay of the path, and I_e is the impairment caused by losses and/or compression. A is the expectation factor.

Based on recommended values in ITU-T Recomm. G.107, the rating R can be defined by

$$R \quad = \quad 94.2 - I_d - I_e$$

Derived by a fitting process in [3], I_d has the following form:

$$I_d \quad = \quad 0.024d + 0.11(d - 177.3)H(d - 177.3)$$

where d is the one-way delay in milliseconds, and H(x) is the Heavyside or step function where H(x) = 0 if x < 0 and 1 otherwise.

Unlike I_d that only depends on the transport network and not on the codecs, I_e is codec dependent and has the following form:

$$I_e \quad = a + b\ln(1 + c\rho)$$

where ρ is the packet loss rate in percentage, while *a*, *b* and *c* are fitting parameters for various codecs, see ITU-T P.833, [4], [5].

The specific values of bitrate (bitr., kb/s), framesize(frs., ms) and *a*, *b* and *c* for different codecs are shown as Table 2. The values for G.729 and G.723.1 are derived in [4], [5].

Fig. 3 shows that the perceived QoS of VoIP with three different codecs are high over time, where the perceived QoS in the in Fig. 3 were estimated with the 97.5 percentile delays.

The most important conclusion from Fig. 3 is that the perceived quality for VoIP is high both for IPv4 and IPv6. Thus, for VoIP, the differences in delay and packet loss between IPv4 and IPv6 do not translate to the perceived quality domain. This result can be explained from the fact that both for IPv4 and IPv6 the measured delay values are below the threshold value of 150 *ms*, which is considered an upper bound for one-way delay for high quality VoIP.

Table 2. Parameters for different codecs

Parameters	G.711	G.729(10ms)	G.723.1
bitr/frsize	64/20	8/10	6.3/30
a	0	10	15
b	30	25.21	36.59
c	15	15	6

5.2 Assessment of perceived quality of MPEG-2 video

In this section we will compare the perceived QoS for MPEG-2 video services over IPv4 and IPv6. For video services there is no standard available mapping network and terminal quality parameters to user perceived quality, expressed as a Mean Opinion Score. Obviously, this is in contrast with VoIP for which this mapping is realized through the E-model, standardized by the ITU-T.

In this paper we use the results of [8] where a relation is established between the perceived QoS of VBR MPEG-2 video MOS_V and the average encoding bit rate *R* (in Mbps), the packet loss ratio *P* and the content type:

$$MOS_v = a \times \left(\frac{R}{b}\right)^{-c} + d + e \times R \times P \quad (1)$$

where the values of the fitting parameters *a, b, c, d* and *e* in (1) depend on the content type.

The model is constructed under the condition that MPEG-2 Transport Stream (TS) packets have a length of 188 Byte, while packet loss occurs according to a two-state Markov process (the so-called Gilbert model) with average burst length 1. Furthermore it is assumed that error concealment is implemented for the MPEG-2 stream.

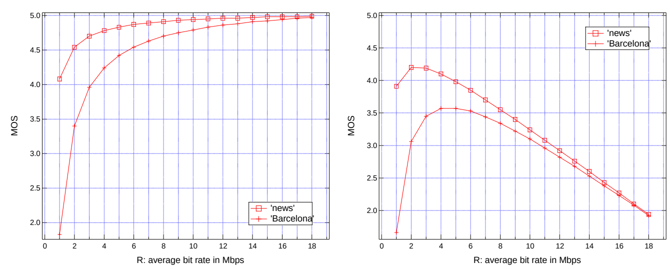
In this paper we consider two content types, referred to in [8] as "news" and "Barcelona". The video sequence "news" is a Head and Shoulder type of sequence and does not contain any high spatio-temporal complexities. On the other hand, the "Barcelona" sequence exhibits a high spatio-temporal complexity, with many scene changes.

The parameters in the model (1) for "news" and "Barcelona" are given in Table 3.

Table 3. Fitting parameters for perceived model (1) for two contents types

	"new"	"Barcelona"
a	-0.025	-0.045
b	60	124.761
c	0.896	0.896
d	5.062	5.225
e	-33.9	-33.9

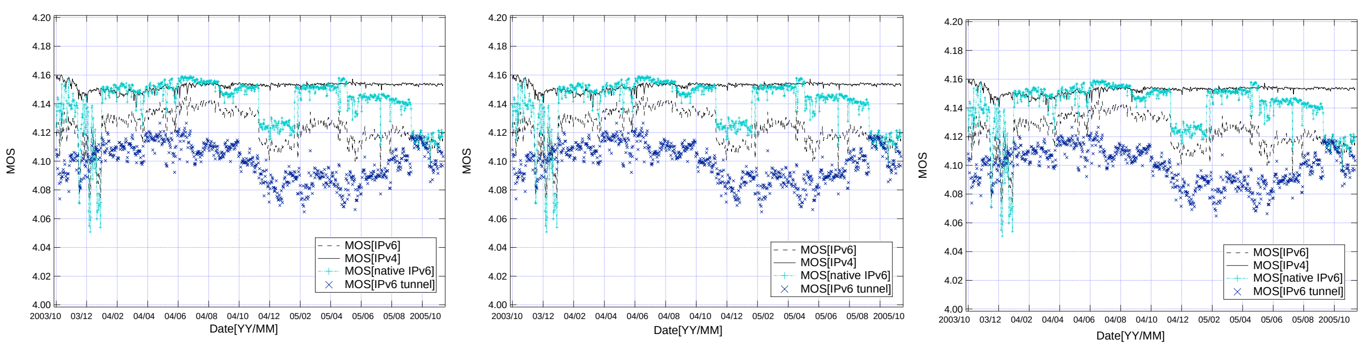
Application of the model (1) is visualized in Fig. 4.



(a) (b)

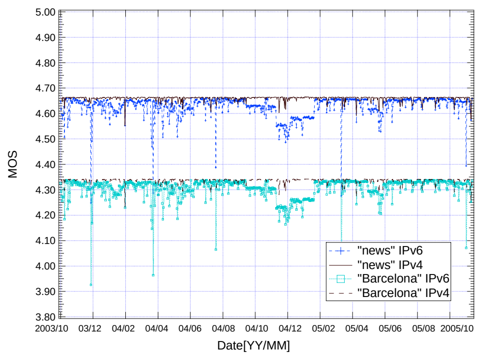
Figure 4. Perceived QoS of MPEG-2 video versus average encoding rate with no packet loss (a) and with packet loss ratio 0.5% (b).

Fig. 4.(a) shows the perceived quality of MPEG-2 video in case of no packet loss. It is clear that the perceptual quality saturates at high bit rates and thus increasing the encoding rate behind this

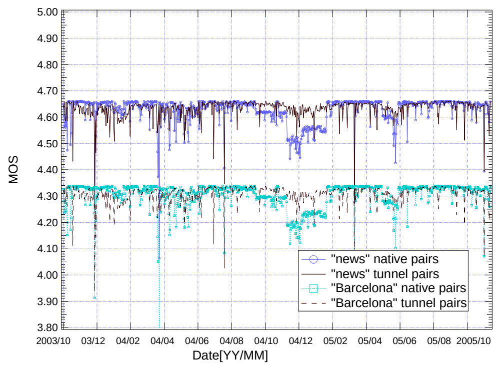


(a) G.711 (b) G.729 (C) G.723.1

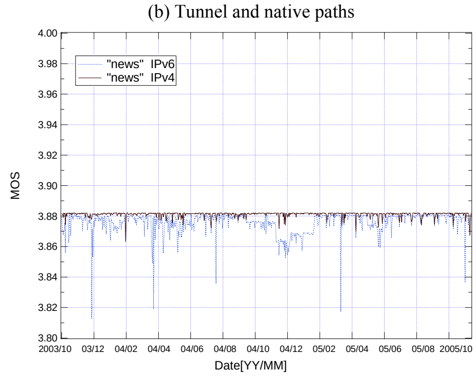
Figure 3. Perceived QoS of VoIP over time for different codecs.



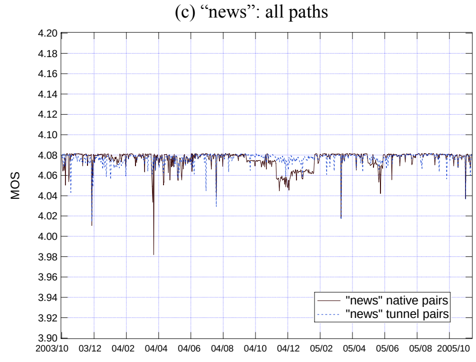
(a) All IPv6 and IPv4 paths



(b) Tunnel and native paths



(c) “news”: all paths



(d) “news”: tunnel and native

Figure 5. Perceived QoS of MPEG-2 video (6 Mpbs) with different models using (a) IPv4 and IPv6 paths; and (b) IPv6 tunnel and native paths; and perceived QoS of MPEG-2 video (1 Mpbs) with "news" model using (c) IPv4 and IPv6 paths; and (d) IPv6 tunnel and native paths.

point will result in a waste of bandwidth.

It is also obvious that for a given average encoding rate the quality of “news” is perceived higher than the quality of “Barcelona”.

Fig. 4.(b) shows the perceived quality of MPEG-2 video in case of a packet loss ratio of 0.5%. It is clear from Fig. 4.(b) that in case of lossy networks there exists an optimal value for the encoding rate R that maximizes the user perceived quality under certain given network conditions. This optimal value depends both on the content type and the packet loss ratio.

Fig. 5.(a) and (b) show that the perceived QoS ofMPEGvideo at a coding rate of 6 Mpbs using IPv6 and IPv4 with "news" and "Barcelona" models over time. Results show that the perceived QoS are relative high (MOS>4) and do not change much over the time. It is also obvious that for a given average encoding rate the quality of "news" is perceived a higher than that of "Barcelona". Moreover, even with the same model, the perceived quality using IPv4 is a littler higher than using IPv6; the perceived quality using native IPv6 is a littler higher than using IPv6 tunnels. Shown as Fig.5.(c) and (d), we observe the similar perceived QoS of MPEG-video at a coding rate of 1 Mpbs using IPv6 and IPv4 with "news" model.

It can be concluded that for Video-over-IP, the differences in delay and packet loss between IPv4 and IPv6 do not translate to the perceived quality domain. This result can be explained from the fact that both for IPv4 and IPv6 the measured packet loss values are very low i.e. less than 0.02%.

5.3 Assessment of TCP performance

In this section we will compare the obtained TCP performance using IPv4 and IPv6.

TCP is the predominant protocol for transporting data over the Internet. For data services using TCP, such as web browsing and file download, there is no standard available mapping network and TCP parameters to user perceived quality. However, it is obvious that for the mentioned services the obtained throughput is an important indicator for quality to users. Therefore in this section we focus on TCP throughput.

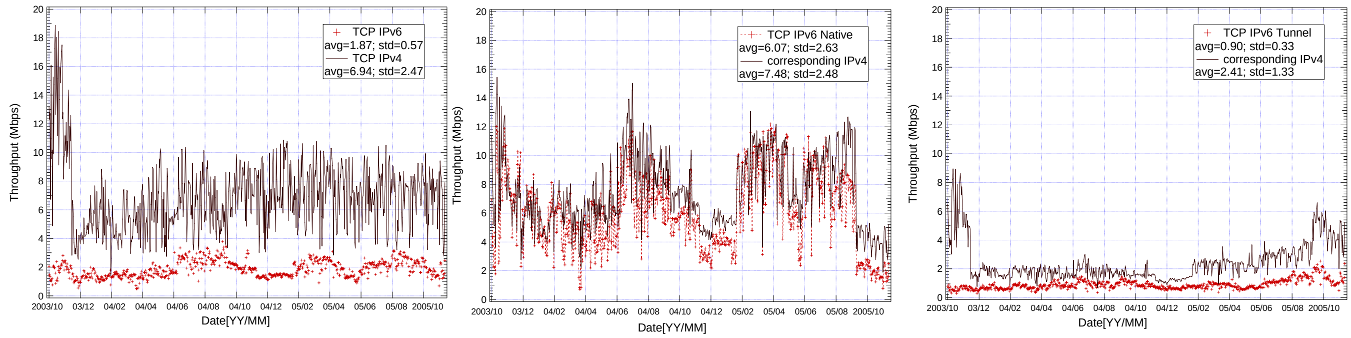
To estimate the TCP throughput we use the well-known formula by Padhye *et al* [6] which gives the steady state throughput R for large file downloads:

$$\min\{\frac{W_{\max}}{RTT}, \frac{MSS}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min(1, 3\sqrt{\frac{3bp}{8}})\, p(1+32p^2)}\} \qquad (2)$$

where Wmax denotes the maximum TCP Window, RTT the round trip time, MSS the maximum segment size, b the number of packets that are acknowledged by an ACK message, T_0 the Retransmission Timer and p the packet loss probability.

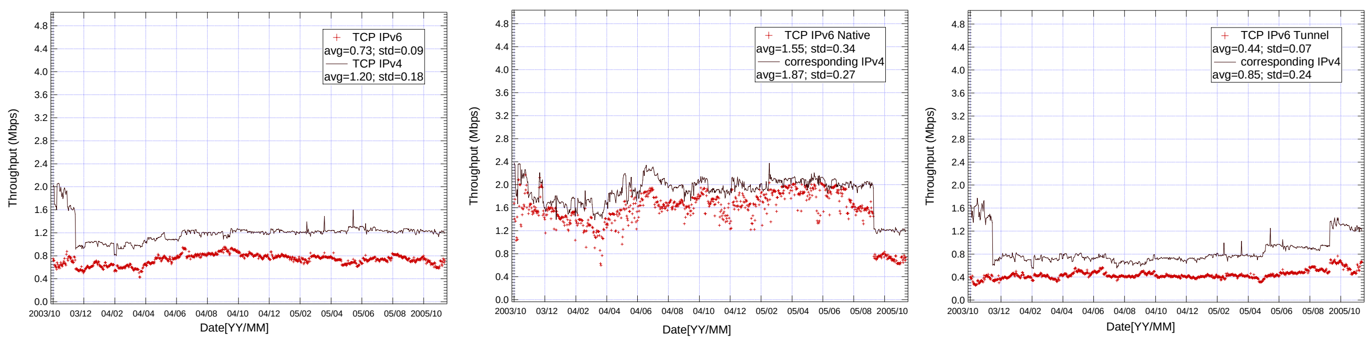
We will apply (2) to estimate TCP throughput for IPv4 and IPv6 under the following assumptions: MSS=1460 bytes; Wmax is sufficiently large, i.e. the throughput is never determined by the maximum window size; RTT equals two times the end-to-end delay; $b = 2$; $T_0 = 4RTT$ (following the heuristic suggested in [7]); and p is the packet loss.

Given Wmax is sufficiently large, Fig. 6 shows the assessment of TCP throughput over time. Fig. 6.(a) shows that the realised TCP throughput for IPv6 is more stable but less high (73% less) than for IPv4 over time. Fig. 6.(b) shows that both native IPv6 and IPv4 counterparts have a variation over time, and the obtained



(a) All IPv6 and IPv4 paths (b) native paths (c) tunnel paths

Figure 6. Assumed Wmax is sufficiently large, assessment of TCP throughput using (a) IPv6 and IPv4 paths; (b) IPv6 native paths and the IPv4 couterparts; and (c) IPv6 tunnel paths and the IPv4 couterparts.



(a) All IPv6 and IPv4 paths (b) native paths (c) tunnel paths

Figure 7. Given Wmax=8192 bytes, assessment of TCP throughput using (a) IPv6 and IPv4 paths; (b) IPv6 native paths and the IPv4 couterparts; and (c) IPv6 tunnel paths and the IPv4 couterparts.

TCP throughput using native IPv6 is close (19% lower) to that of IPv4. Fig. 6.(c) shows that the realised TCP throughput of IPv6 tunnels and IPv4 counterparts are stable over time, and the perceived TCP throughput of IPv6 tunnels is much less (63% less) than that of IPv4. Fig. 6.(b)(c) show TCP throughput both for native IPv6 and IPv6 tunnels. On average, the realized TCP throughput of native IPv6 is much higher (674% higher) than that of IPv6 tunnels.

In the Windows operating systems like Windows 2000, the default value of the TCP Window Size can be set as the larger of four times the maximum TCP data size on the network or 8192 rounded up to an even multiple of the network TCP data size. This parameter determines the maximum TCP receive window size offered by the system. The receive window specifies the number of bytes a sender may transmit without receiving an acknowledgment. In general, larger receive windows will improve performance over high delay networks. For highest efficiency, the receive window should be an even multiple of the TCP Maximum Segment Size (MSS).

Given Wmax = 8192, Fig.7 shows the assessment of TCP throughput over time. Fig. 7.(a) shows that the TCP throughput for IPv6 is quite stable but less high (39% less) than for IPv4. Fig. 7.(b) shows that both native IPv6 and IPv4 counterparts did not change over time, and the obtained TCP throughput using native IPv6 is close (17% lower) to that of IPv4. Fig. 7.(c) shows that the realized TCP throughput of IPv6 tunnels and IPv4 counterparts are quite stable over time, and the TCP throughput of IPv6 tunnels is much less (48% less) than that of IPv4.Fig. 7.(b)(c) show TCP throughput both for native IPv6 and IPv6 tunnels.

On average, the realised TCP throughput of native IPv6 is much higher (252% higher) than that of IPv6 tunnels. In reality, Wmax is often determined by access rate. Let us assume a (downlink) maximum access rate of 5 Mbps. For this situation, which typically corresponds with an ADSL2+ access, the corresponding function for throughput for (2) has the following form:

$$\min\{5, \frac{MSS}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min(1, 3\sqrt{\frac{3bp}{8}})p(1+32p^2)}\} \tag{3}$$

Given ASDL2+ with a rate 5 Mbps, Fig. 8 shows the assessment of TCP throughput over time. Fig. 8.(a) shows that the realised TCP throughput using IPv4 is only limited by the ADSL2+ speed, while for IPv6 the realised throughput is about 16% less. Fig. 8.(b) shows that obtained TCP throughput using both native IPv6 and IPv4 counterparts is limited by the ADSL2+ speed, and did not change a lot over time. Fig. 8.(c) shows that the realised TCP throughput using IPv6 tunnels and IPv4 have a variation over time, and the obtained TCP throughput of IPv6 tunnels is less (62% less) than that of IPv4, however, it has been steadily increasing in the end of 2005. Fig. 8.(b)(c) show that the TCP throughput using native IPv6 is almost only limited by the ADSL2+ speed, and that for IPv6 tunnels the throughput is less stable. On average, the realised TCP throughput using IPv6 tunnel is lower (79% less) than for native IPv6. The most important conclusion from the observations above is that for applications based upon TCP, the differences in delay and packet loss between IPv4 and IPv6 have a strong impact on the realised throughput. This result can be explained from the fact that TCP throughput is very sensitive with respect to Round Trip Time. Based on the above measurements, we conclude that given Wmax is sufficient large or Wmax = 8192, the realised TCP throughput for IPv6 and IPv4 are relative high. When the bottleneck speed is in the access link, such as for ADSL2+, the obtained TCP throughput using native IPv6 and IPv4 are mainly limited by the access speed. In general, the best TCP throughput performance is obtained by using IPv4, then followed by native IPv6. The worse performance is obtained by using IPv6 tunnels.

6. CONCLUSION

In this paper, we have presented a detailed measurement study and an analysis of the delay and loss evolution over two years in an IPv6 network. We believe that this study is a contribution to the ongoing effort to gain insight into the behavior of large-scale IPv6 networks. We have shown that since October 2003, the median IPv6 delay has become smaller; it started at about 120 ms on Oct. 2003 and decreased to about 55 ms on Oct. 2005.

We have shown that IPv6 network has a higher delay evolution than IPv4. Furthermore we have shown that both IPv6 and IPv4 have a relative small packet loss. The most important conclusion for the perceived quality of services running over IPv6 networks is: for VoIP and Video-over-IP, the differences in delay and packet loss between IPv4 and IPv6 do not translate to the perceived quality domain; for applications based upon TCP, the differences in delay and packet loss between IPv4 and IPv6 have a strong impact on the realised throughput.

7. REFERENCES

[1] Adam, Y., B. Fillinger, I. Astic, A. Lahmadi, and P. Brigant, "Deployment and Test of IPv6 Services in the VTHD Network", IEEE Communications Magazine, vol. 42, no. 1, January 2004

[2] Cho, K., M. Luckie, and B. Huffaker, "Identifying IPv6 Network Problems in the Dual-Stack World", SIGCOMM'04 Network Troubleshooting Workshop, Oregon, Sep. 2004

[3] Cole, R.G., and J. H. Rosenbluth, "Voice over IP Performance Monitoring", SIGCOMM Computer Communication Rev., 31(2):9-24, 2001.

[4] Ding, L., and R. A. Goubran, "Speech quality prediction in voip using the extended e-model", IEEE GLOBECOM, Dec. 2003.

[5] Ding, L., and R. Goubran, "Assessment of effects of packet loss on speech quality in VoIP", Proc. of the 2nd IEEE HAVE 2003, Canada, September 2003, pp. 49-54

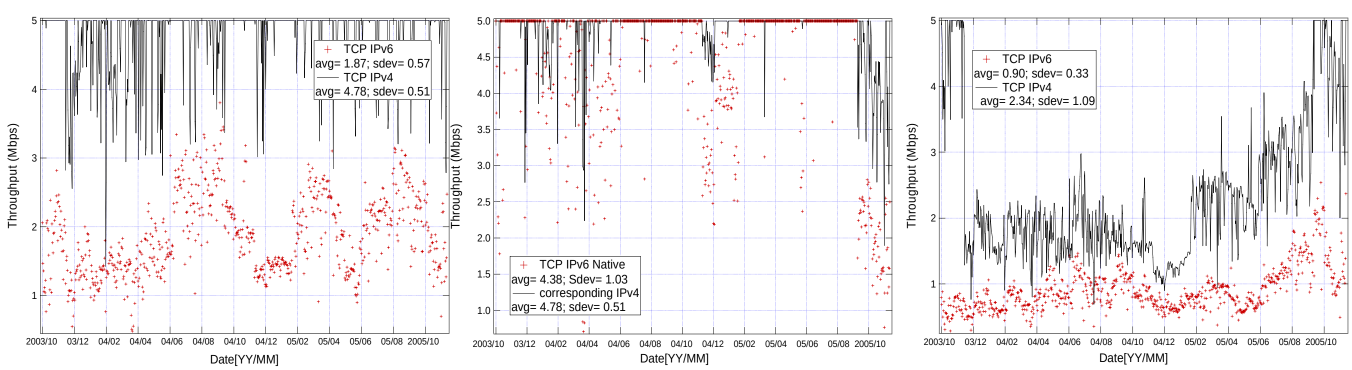
[6] Padhye, J., V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A Simple Model and its Empirical Validation", IEEE/ACM Transactions on Networking, 8(2): 133-145 (April), 2000

[7] Paxson, V., and M. Allman, "Computing TCP's Retransmission Timer", Internet Archives RFC 2988, November 2000

[8] Verscheure, O., P. Frossard, and M. Hamdi, "MPEG-2 Video Services over Packet Networks: Joint Effect of Encoding Rate and Data Loss on User-Oriented QoS", NOSSDAV 98, Cambridge, July 1998

[9] Yajnik, M., S. Moon, J. Kurose, and D. Townsley, "Measurement and Modeling of the Temporal Dependence in Packet Loss," IEEE INFOCOM'99, March 1999.

[10] Zhou, X., M. Jacobsson, H. Uijterwaal, and P. Van Mieghem, "IPv6 Delay and Loss Performance Evolution", submitted to International Journal of Computer Networks, 2006



(a) All IPv6 and IPv4 paths　　(b) native paths　　(c) tunnel paths

Figure 8. Given ADSL2+ with a speed 5 Mbps, assessment of TCP throughput using (a) IPv6 and IPv4 paths; (b) IPv6 native paths and the IPv4 couterparts; and (c) IPv6 tunnel paths and the IPv4 couterparts.