The Utility of Characterizing the Buffer of Network Devices in order to Improve Real-time Interactive Services

Luis Sequeira, Idelkys Quintana, Jose Saldana, Luis Casadesus, Julián Fernández-Navajas, José Ruiz-Mas
Communications Technology Group (GTC)-Aragón Inst. of Engineering Research (I3A)
Dpt. IEC. Ada Byron Building. EINA. Univ. Zaragoza
50018 Zaragoza, Spain
Email: {sequeira, idelkysq, jsaldana, luis.casadesus, navajas, jruiz}@unizar.es

ABSTRACT

The behaviour of the routers' buffer is of primary importance when studying network traffic, since it may modify some characteristics, as delay or jitter, and may also drop packets. As a consequence, the characterization of this buffer is interesting, especially when real-time flows of interactive services are being transmitted. If the buffer characteristics are known, then different techniques can be used so as to adapt the traffic: multiplexing a number of small packets into a big one or fragmentation.

This work presents a study of how to determine the technical and functional characteristics of the buffer of a certain device (e.g. size, input and output rate and others characteristics related to its behaviour), or even in a remote node. Two different methodologies are considered, and tested on two real scenarios which have been implemented; real measurements permit the estimation of the buffer size, and the input and output rates, when we have physical or remote access to the "System Under Test". In case of having physical access, the maximum number of packets in the queue can be determined by counting. In contrast, if the node is remote, its buffer size can be estimated. We have obtained accurate results in wired and wireless networks.

Categories and Subject Descriptors

H.1 [**Models and Principles**]: General; I.6 [**Computing Methodologies**]: Simulation and Modeling

General Terms

Algorithms, Experimentation, Measurement.

Keywords

Buffer size, queueing, unattended measurements, P2P-TV, streaming.

LANC'12, October 4-5, 2012, Medellín, Colombia

1. INTRODUCTION

The large increase in the number of users and new multimedia services over Internet (e.g. P2P-TV streaming, VoIP, online games) generate a significant amount of network traffic [1]. Moreover, these services and applications generate packets which size can vary from tens of bytes as VoIP or online games, to bigger sizes in others cases, increasing information efficiency respect to headers. While some services inject traffic with constant bit rate in order to provide a certain Quality of Service (QoS) and a better user's experience [2, 3], there are other applications that generate bursty traffic with different amount of frames [4].

These interactive real-time services, as IPTV, have an important impact on network traffic, since the bandwidth requirements derived from sending a unicast stream to each user would become prohibitively expensive [5]. P2P-TV is a practical solution to this scalability problems, since it decreases the bandwidth and resources cost [6, 7]. However, it is necesary to know and to evaluate in more detail the impact of this traffic and its influence on network infraestructure.

The growth in the demand of high amounts of data by Internet users from interactive real-time services, in conjunction with the nowadays existing complex network architectures, produces critical bottlenecks in some network points. This mainly happens in access networks, because capabilities are lower than in the backbone; although these critical points can also appear in high-performance networks. In these points, the main cause of packet loss is the discarding of packets in router queues. So the implementation of router buffers in conjunction with scheduling policies, are of primary importance in order to ensure the correct delivery of the traffic of interactive real-time services.

Access networks normally use mid and low-end routers, which do not usually implement advanced traffic management techniques. However, they use buffers as a traffic regulator mechanism, so buffer size becomes an interesting design parameter. Most Internet routers use FIFO drop-tail buffers [8, 9] but there exist another techniques to manage these buffers, e.g. Random Early Detection (RED). These techniques, in conjunction with buffer size, mainly define the buffer behaviour, and therefore, how the traffic is affected by it [10, 11].

As a consequence, if the size of the buffer and its behaviour are known, some techniques to manage packet size can be used so as to improve link utilization: multiplexing a number of small packets into a big one, or fragmentation. For these situations, our group is currently working on the

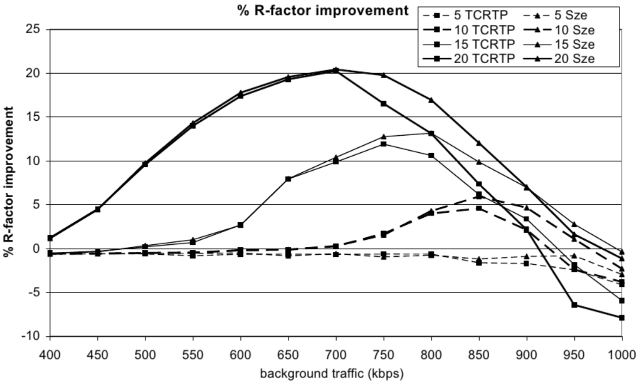


Figure 1: R-factor improvement with different multiplexed calls [12].

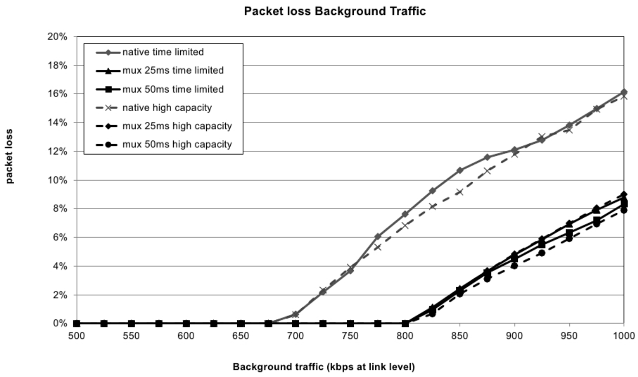


Figure 2: Packet loss for background traffic for tow buffers [13].

development of a tool able to discovery some characteristics of the buffer and its behaviour. The idea is to permit measurements not only when we have physical access to "System Under Test" but also in the case of only having remote access.

The paper is organized as follows: section II discusses the influence of the buffer in different real-time services, in order to highlight the need of knowing the buffer behaviour. In this sense, section III presents the test methodology. Its aim is to find the buffer's technical and functional characteristics. The next section covers the obtained experimental results. The paper ends with the conclusions.

2. INFLUENCE OF THE BUFFER IN INTERACTIVE REAL-TIME SERVICES

In the literature, we can find different studies related to the influence of the buffer on different services and applications. These works show how QoS is affected by the buffer behaviour, size and scheduling policies. In these cases, knowing the technical and functional characteristics of this device becomes critical. This knowledge can be useful for applications and services in order to make correct decisions in the way the traffic is generated, using different optimized management packets techniques.

In [12] the behaviour of the buffer size was studied on VoIP, where three different router's buffer policies (dedicated, big and time-limited buffer) were tested, also using two multiplexing schemes. Router's buffer policies cause different behaviour in packet loss, as well as R-factor (see Fig. 1). The authors show a multiplexing method of VoIP flows, thus reducing bandwidth but at the cost of increasing packet size, and this will have an influence on packet loss, depending on the implementation and buffer size of the router. In this case the VoIP native traffic showed a good behaviour when using a small buffer, as small packets have less probability of being discarded than big ones.

In [13] the authors present study of the influence of a multiplexing method on the parameters that define the subjective quality of online games: mainly delay, jitter and packet loss. The work considers two buffer implementations, each one with two buffer sizes, in order to study the relationship between router's buffer and multiplexing on subjective quality. The multiplexed traffic of a number of users shared the same access link. The results show that small buffers present better characteristics in order to maintain delay and jitter in adequate levels, at the cost of increasing packet loss (see Fig. 2). In addition, buffers whose size is measured in packets also increase packets loss.

In this context, our group work-in-progress shows that P2P-TV can also be influenced by buffer behaviour. The next section presents a summary of our preliminary findings.

2.1 P2P-TV traffic

In recent years, several studies have characterized P2P video streaming application and have measured their impact in the communication networks traffic [6, 7, 14, 15]. So, in [14] is presented a traffic characterization of major P2P-TV applications and they concluded that each application generates it own traffic pattern, but they have in common that the traffic consists in a mixture of small packets (signalling packets) and large packets (video packets).

One of this applications is SOPCast, it relies on UDP traffic and it is revealed that this application faces a high overhead, about 60% of packets correspond to signalling, and the other 40% correspond to video data packets [16]. For this application we can find a summary of packets lengths and their corresponding functionalities (see Table 1).

Many access network devices are designed for bulk data transfers [17], such as mail, web or FTP services. However, the P2P video streaming applications generate a high rate of small packets, so the routers may experience problems in order to manage all the packets and their processing capacity can become a bottleneck if they have to manage too many packets per second. Furthermore, the generation of large number of small packets [16] may cause that buffer penalizes the video packets distribution and consequently peer's behaviour within P2P structure is not the expected. Buffer policies and its size could be the cause of the packet loss.

In this context, our group has been performed some tests related with the effect of P2P-TV traffic generated by SOPCast on router buffer, taking into account the packet size and different Internet access technologies. Below, the testbed is described and the obtained results are presented.

2.1.1 Testbed

Two real P2P-TV traces were used from NAPA-WINE project [18]. These traces use SOPCast over LAN and ADSL. In this case, 44 peers distributed over 4 countries were watch-

Table 1: Description of generated UDP packets by SOPCast [15].

$Type$	$Size\ (bytes)$	$Funcionality$
$Video\ Packet$	1320^*	$Maximum\ size\ of\ the\ video\ packets$
$Control\ Packet$	52	$HELLO\ packet\ to\ initiate\ link\ connections$
	80	$Confirmation\ on\ receiving\ the\ HELLO\ packet$
	28	$Acknowledgement$
	42	$Keep-alive\ message\ with\ neighbors$
	46	$Video\ requesting\ packet$

$^*There\ are\ also\ fragments\ of\ video\ packets.$

ing the same channel during 1 hour at the same time. Fig. 3 shows the scenario used for this tests. These simulations use P2P traffic and the background traffic share the same access link. Traces are sent following exactly the times obtained and original sizes of the capture, so a traffic model have not been necessary. The size distribution of background traffic is: 50% of the packets are of 40 *bytes*, 10% of 576 *bytes* and 40% of 1500 *bytes* [19].

We have used two buffers implementations: the first one defines the size in bytes and three different sizes have been chosen: 10, 100 y 1000 *Kbytes*. The second defines the size in number of packets, also different lengths have been used: 27, 270, 2700 *packets*. In this test, we have used different values for the upload bandwidth: 512, 1024 y 2048 *Kbps*, commonly used for Internet access today.

2.1.2 Traffic analysis

Fig. 4 and 5 show the results for both traces. It should be noted that packet loss in the buffers limited in number of packets are well above those buffer of limited in bytes. This is due to the large amount of packets per seconds that SOPCast generates and take a place in the buffer regardless of size.

When the buffer size is limited in bytes longer packets are principally discarded because small packets fill up the buffer leaving no enough space for larger packets, so packet loss increases (see Fig. 4 and 5). Packet loss causes that a peer does not contribute properly in video distribution content to other peers in P2P network. In this case, background services can be affected such as FTP, HTTP or any other that generates large packet traffic. Results show that buffer behaviour is very similar in both used technologies.

On the other hand, if the buffer size is limited in number of packets, it discards signalling packets in the same extent as video packets. Small packets make that buffer queue fills up faster and the its policies penalizes both the small packets and large packets, then the upload and download video are impaired (see Fig. 4 and 5). Again, background services can be affect and buffer behaviour is similar in LAN and ADSL.

In addition, P2P video distribution would be affected by small packets generated by applications that use the same link to Internet access, such as VoIP and online games.

3. TEST ENVIRONMENT AND METHODOLOGY

In this section, test environment and studied buffer characteristics will be presented. Also, the methodology used to describe buffer behaviour, size, input and output rate is going to be explained.

3.1 Buffer: types and behaviour

Router buffer size is not a parameter appearing in the technical specifications that manufacturers provide with their devices. However, this design characteristic is important when planning a network. The reason for this is that there is a relationship between router buffer size and link utilization, since an excessive amount of memory would generate a significant latency increment when the buffer is full. On the other hand, a very small amount of memory in the buffer will increase packet loss in congestion time. As a consequence, the knowledge of the buffer behaviour is an interesting parameter which can be considered when trying to improve link utilization. In the last years, there are many studies related to the buffer size issues but they are principally approached in backbone routers and TCP flows.

Until a few years the researchers accepted the BDP (Bandwidth Delay Product) [20] as a method to obtain the buffer size needed at a router's output interface. This rule was proposed in 1994 [21] and it is given by $B = C \times RTT$, where B is the buffer size, RTT is the average round-trip time and C the capacity of the router's network interface. In [22], proposed a reduced buffer size by dividing BDP by the square root of the number of used TCP flows $B = C \times RTT/\sqrt{N}$. This model was called *small buffer* [23].

In [24] it was suggested the use of even smaller buffers, called *tiny buffers*, considering a buffer size of some tens of packets. However, the use of this model presents a discarding packet provability of $10\% - 20\%$. In addiction, it has also been observed in the literature that the buffer size is measured in different ways: e.g., in [25] the routers of two manufactures are compared, and one gives the information in packets, whereas the other one measures it in milliseconds, which is equivalent in the practice to measure it in bytes.

3.2 Test scenarios

The scheme used for the tests is shown in Fig. 6. This scenario permits to analyse a "System Under Test" (SUT from now), which may be either a device or a network (switch ethernet, WiFi link, cable modem, etc.). Two different scenarios are going to be considered: in the first case there is physical access to the SUT, and in the other the access is

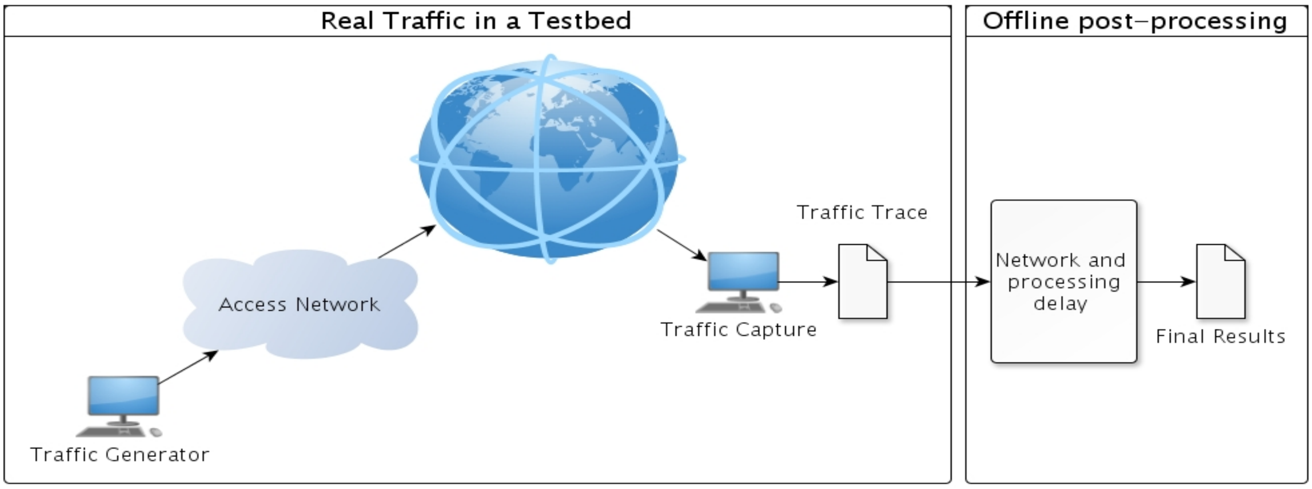
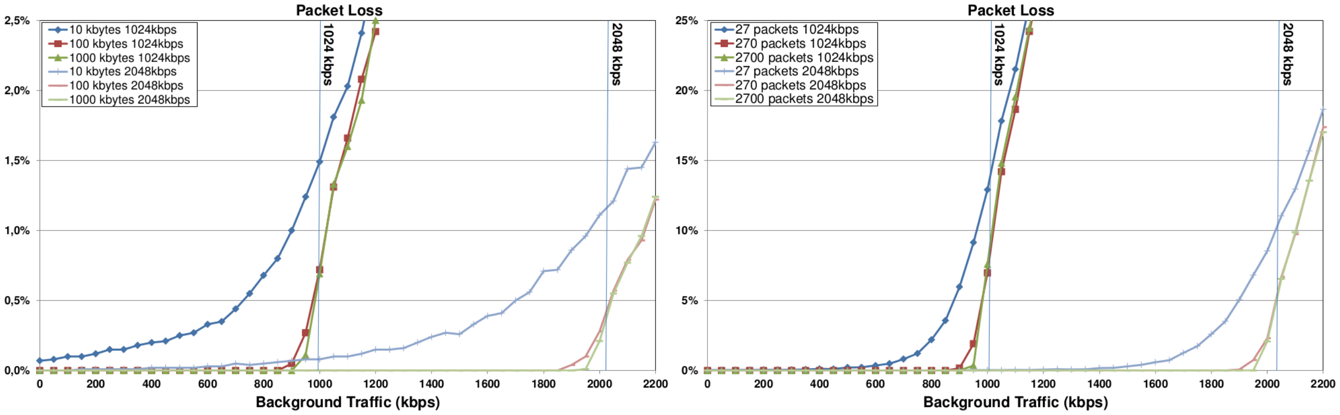


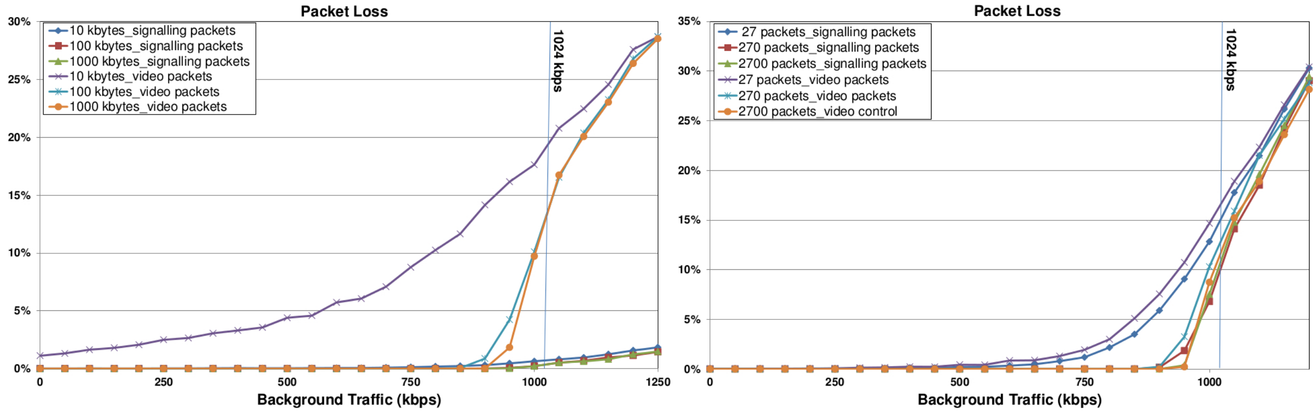
Figure 3: Test scenario.



(a) Buffer size limited in bytes.

(b) Buffer size limited in packets.

Figure 4: Buffer packet loss in ADSL.



(a) Buffer size limited in bytes.

(b) Buffer size limited in packets.

Figure 5: Packet loss according packet size in ADSL.

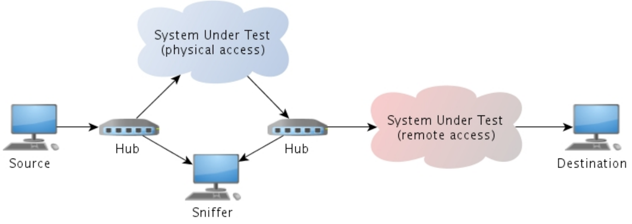


Figure 6: Topology used for test.

remote. The proposed methodology permits to determinate certain SUT characteristics when physical access is guaranteed. Also, accurate results have been obtained with remote access.

In this scenario, a pair of probes have been allocated in the ends of the communication. Traffic is sent from a *source*, and arrives to *destination* traversing the SUT. The test is based on the sending of a burst of UDP packets from the *source* to the *destination* machine, so as to produce a buffer overflow in the SUT. When there is physical access a *sniffer* is used to capture the traffic at the input and output of the SUT. On the other hand, unattended data capture is managed in *destination* equipment in the case of remote access.

3.3 Methodology

We will use two methodologies to estimate buffer behaviour, size, thresholds and output rate. These methods are used depending on the type of capture in the test environment: with physical or remote access. In the case of having physical access, *sniffer* captures two files at the input and output of the SUT, in remote access only one file is captured in *destination* equipment. The first method is exact, since captures give the values for all needed variables, in the second method estimations algorithms are needed. Therefore, the first one has been used as a reference point to develop the algorithms. The methods are described as follow:

3.3.1 Buffer size with physical access

Both files are processed with a *shell script* to calculate packet delay, packet loss, inter packet time, input and output buffer rate and filling buffer rate. Buffer size is determined, for each packet in output captured file, counting packets between input and output time at the buffer (see Fig. 7).

3.3.2 Buffer size with remote access

Fig. 8 shows the relationship between sent and received packet times. Let R_{in} and R_{out} the input and output rates of the buffer, respectivelly. We define R_{fill} as the rate in which the buffer fills when R_{in} is bigger than R_{out} ($R_{fill} = R_{in} - R_{out}$). L_{buffer} is the size of the buffer in bytes. First, T_r can be obtained from output capture file and it depends of buffer size (L_{buffer}). T_r is the time between the arrival of the first packet until the first dropped packet is detected, also it is equal to the filling time plus the time that takes a packet to traverse the buffer when the queue is full. It can be obtained as: $T_r = \frac{L_{buffer}}{R_{fill}} + \frac{L_{buffer}}{R_{out}}$ and solving for buffer length:

$$L_{buffer} = \frac{T_r}{\frac{1}{R_{in}-R_{out}} + \frac{1}{R_{out}}} \tag{1}$$

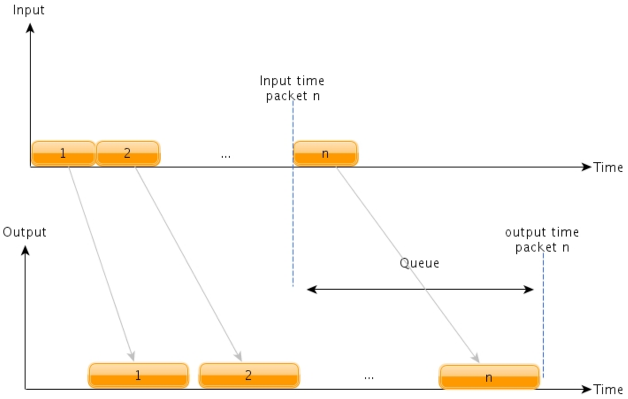


Figure 7: Estimating packets in queuing.

The output rate can be easily determined, because the remote capture includes the arrival time for each packet and packet length is known, additionally filling rate is the difference between input and output rate. Input rate is not as trivial as output rate, because we only have the *destination* capture. Let τ the time interval to estimate R_{in} and R_{out}. It is the time that takes to send the received and dropped packets. Input rate can be obtained as follow: τ can be exactly measured and it will have the same value in both extremes. In this case, when *destination* receives n consecutive packets, the *source* has sent $n + m$ packets (where m is the number of dropped packets). However, dropped packets can be known because all the packets have a unique identifier. With this information, output and input rate can be estimated only from the data contained in the *destination* capture, using the following expression:

$$R_{in} = \frac{n_{tx} + m_{tx}}{\tau} \times packet_{size} \tag{2}$$

Now, buffer length can be obtained substituting the equation 2 in 1.

4. CHARACTERIZATION OF THE BUFFER WITH PHYSICAL AND REMOTE ACCESS

In order to illustrate the proposed methodology, real tests have been deployed in a testbed, which scheme is shown in Fig. 6 and results are analysed based on procedures shown in Fig. 7 and 8. Real machines have been used (Linux kernel $2.6.38-7$, Atheros $AR9287$ wireless network adapter, $Intel^{\circledR}$ $Core^{TM}$ $i3$ CPU 2.4 GHz) in a laboratory environment, in order to identify the buffer behaviour of different devices.

ETG (E2E Traffic Generator) is an application developed by our group [26] and it was used in this work to generate the IP flows. It allows us to calculate objective quality parameters (delay, jitter and loss rate) and also subjective ones (R factor, MOS). It is capable of establishing E2E communications between hosts, by sending and receiving multiple UDP traffic bursts. It generates one-way and round trip traffic and captures it at the ends of the communication. UDP flows have been used with different packet size because it is the type of traffic used by SOPCast (Table 1).

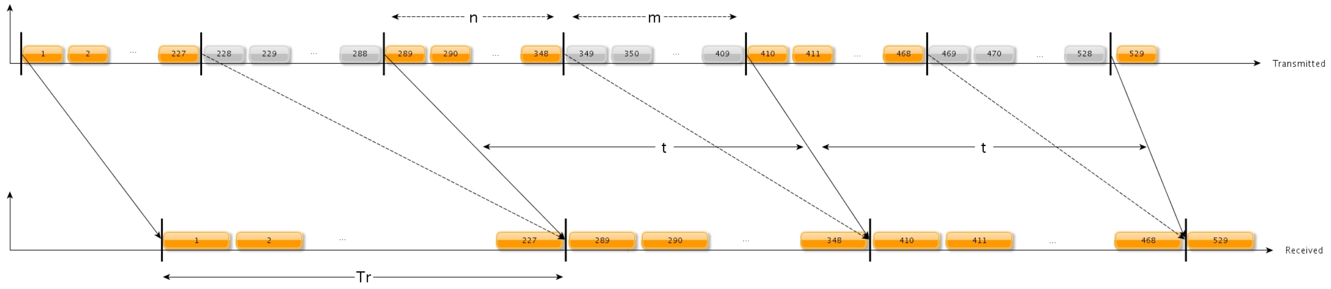


Figure 8: Estimating packets in queue in remote access.

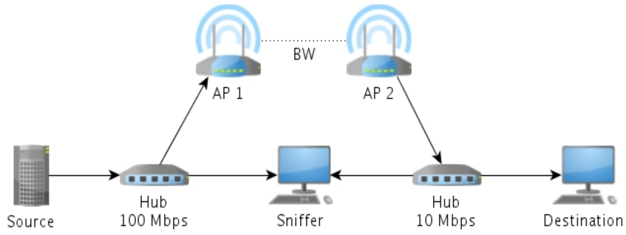


Figure 9: First scenario: Estimating buffer size in a wireless network.

4.1 Real scenarios

We have studied two different scenarios, in both cases a host sends traffic to a destination through a network or a device. A sniffer is included at the best location for making captures. ETG was used to generate UDP flows to flood the buffer, and also to automate *tcpdump* captures in the ends of the link. Flows were sent from *source* to *destination*. Two hubs with different bandwidths were used to create a bottleneck which has to be measured. Flow control options in the switch are disabled.

4.1.1 Wireless network

In the first case, a link between two access points (Linksys WAP54G) has been analized, trying to estimate the buffer size of the first one ($AP1$). Fig. 9 shows the used topology: both hosts are connected to the hubs using 100 $Mbps$ and 10 $Mbps$ links respectively. UDP flows were sent from *source* to *destination*, with the aim of obtaining the buffer size of $AP1$.

4.1.2 Wired network

For the second scenario, Fig. 10 shows used topology, a typical home network accesses to Internet through cable modem link. This network is accessed by a host from a different network across the Internet. This scenario permits access to the *switch*, so the measurements can estimate switch's buffer size.

4.2 Measures with physical and remote access

The presented measures in this section show a comparison between the proposed methods to determinate the buffer size with physical and remote access. The tests were made with the aim to identify particular behaviour in buffers and its size. Different saturation rates have been used in the tests to assess the optimum rate. On the other hand, some technologies present variations in their transmission rate to counteract the channel conditions. Due to this, different bandwidths have been configured when technology allows it, to value the accuracy of the algorithms. Packets with three different sizes have been used in UDP flows to overflow the buffer with the aim to determinate if the buffer size is defined en number of packets or bytes.

4.2.1 First scenario

In this wireless scenario we have compared the two methods and the results are shown in Table 2. First, we observed a particular buffer behaviour: when the buffer is completely full no more packets are accepted, it will be called *upper threshold*. In this moment, the buffer does not accept new packets until a certain amount of memory is available, it will be called *lower threshold*, and in this moment the filling process begins again. This can be seen in Fig. 11.

Although this behaviour has some similarities with Random Early Detection (RED), it is not the same: when RED is used, if the buffer is almost empty, all incoming packets are accepted but when the number of enqueued packets grows, the probability for dropping an incoming packet grows too. Finally, if the buffer is full, all incoming packets are dropped. On the other hand, the observed buffer behaviour does not use any dropping probability: if the buffer is full, it does not accept new packets until a certain amount of memory is available.

Furthermore, the variations of the output rate in wireless network (see Table 3) generate an error growth. The reason for this is that the WiFi access point switches from higher to lower speeds depending on the state of the radio channel. Each test was repeated for three different packet size and the buffer size is the same, so we conclude that the buffer size is defined in number of packets. Also, most accurate results have been observed using bigger packets.

The presented results are the ones obtained using packets of 1500 bytes, since they are the most accurate. It can be seen that *method 1* is the most accurate estimation so it has been used to compare with *method 2*.

4.2.2 Second scenario

In this case, the two proposed methods were compared in the face of input rate variations of the buffer in Ethernet, see Table 4. In this new scenario the same buffer behaviour has been observed: if the buffer is completely full no more packets are accepted until a certain amount of memory is available.

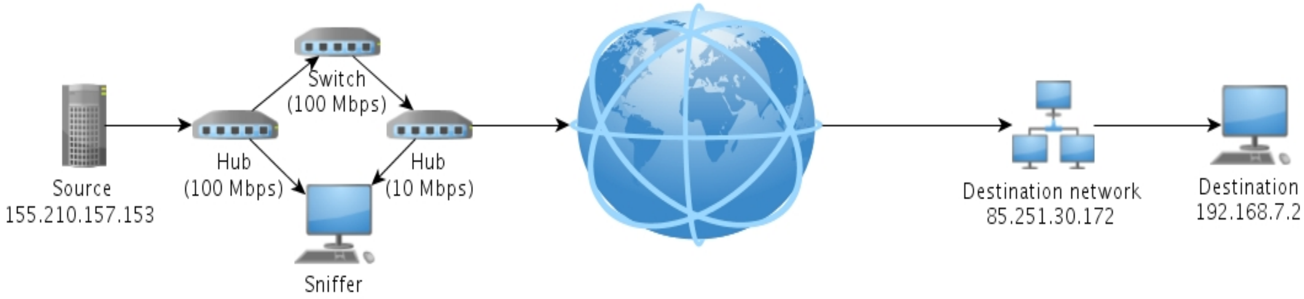


Figure 10: Second scenario: Estimating buffer size in a wired network.

Table 2: Wifi access point buffer size (in packets) for different bandwidth.

Access point bandwidth		*Method 1*		*Method 2*		*Method 2 error (%)*	
Mbps		*LT*	*UT*	*LT*	*UT*	*LT*	*UT*
$Wi-Fi$	1	30	55	30	55	0	0
	2	30	55	30	55	0	0
	5.5	30	55	30	55	0	0
	11	30	55	32	53	6.67	3.63
	24	30	55	33	52	10	5.45
	54	30	55	36	59	20	7.27

LT: Stands for lower threshold.

UT: Stands for upper threshold.

Table 3: Observed variations in buffer output rate for different bandwidth.

Access point rate	54 *Mbps*	24 *Mbps*	11 *Mbps*	5.5 *Mbps*	2 *Mbps*	1 *Mbps*
Minimum(Mbps)	10.88	13.7	5.75	2.29	1.24	0.65
Maximum(Mbps)	28.36	16.84	6	3.13	1.41	0.65

Table 4: Ethernet buffer size for different bandwidth.

Bandwidth		*Method 1*		*Method 2*		*Method 2 error (%)*	
Mbps		*LT*	*UT*	*LT*	*UT*	*LT*	*UT*
Ethernet (physical)	20	85	115	83	113	2.35	1.74
	30	85	115	84	114	1.17	0.87
	40	85	115	84	115	1.17	0
Ethernet (remote)	20	85^*	115^*	80	111	5.89	3.48
	30	85^*	115^*	83	113	2.35	1.74
	40	85^*	115^*	84	115	1.17	0

LT : Stands for lower threshold.

UT : Stands for upper threshold.

**These values were calculate as physical access to compare method 2.*

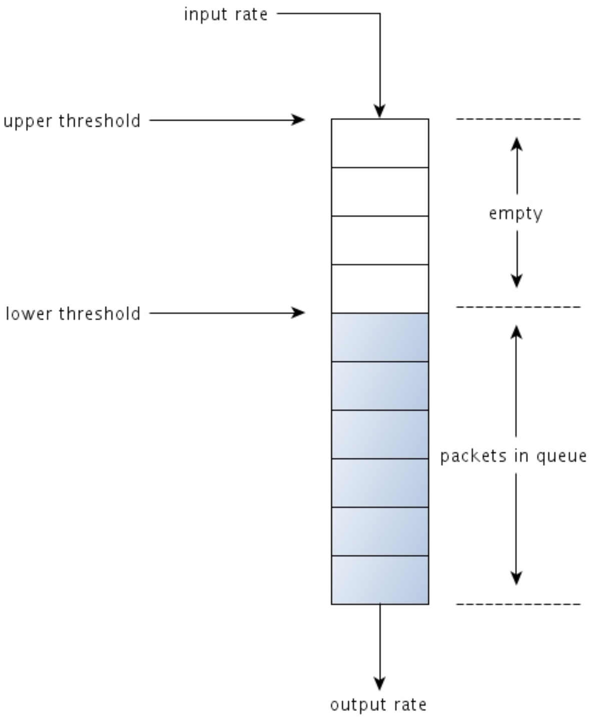


Figure 11: Buffer characteristics.

On the other hand, although there is no variations in output rate, three different input bandwidth have been used to cause a overflow in the buffer, and this most accuracy results were obtained when input rate grows. Again, buffer size in defined in number of packets because results are the same for different packet size.

If we look at the *physical access* results (first three rows of the table), we see that the accuracy of the buffer size estimation using the *method 2* is high. In addition, the error decreases as input rate increases.

Regarding *remote access* results, it should be noticed that the results are less accurate but they are still acceptable. As integer values for the buffer capacity are used, in some cases the results are exactly the same than the ones of *method 1*, so the obtained value of the error is null.

5. CONCLUSIONS

This paper presents two methods to analyse the technical and functional characteristics of commercial buffers of different devices, or even networks. This characterization is important for interactive real-time services, taking into account that the buffer may modify traffic characteristics in different services: as e.g. in P2P-TV peers do not contribute to video distribution and may also drop packets. Also, packet size has to be considered because some buffer policies penalizes the large packets.

The methodology can be used if there is physical access to the "System Under Test", but it is also useful, with certain limitations, for measuring a remote system. Tests using commercial devices have been deployed in two different scenarios, using wired and wireless networks. A particular buffer behaviour has been observed for a device: once the buffer is full, it does not accept new packets until a certain space is again available.

The results show that accurate results of the buffer size can be obtained when there is physical access to the "System Under Test". In case of having no direct access to the system, an acceptable estimation can also be obtained if the input rate is sufficient high. In this case, big packets have to be used for the tests. This confirms the good performance of the proposed algorithms.

6. ACKNOWLEDGMENTS

This work has been partially financed by CPUFLIPI Project (MICINN TIN2010-17298), MBACToIP Project, of Aragon I+D Agency, Ibercaja Obra Social, Project of Cátedra Telefónica, Univ. of Zaragoza, Banco Santander and Fundación Carolina.

7. REFERENCES

[1] Gao Huang, Meng Ye, and Long Cheng. Modeling system performance in mmorpg. In *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE*, pages 512 – 518, nov.-3 dec. 2004.

[2] Alfio Lombardo, Giacomo Morabito, and Giovanni Schembra. Statistical traffic modeling and guaranteed service disciplines: a performance evaluation paradigm. *Computer Networks*, 36(5-6):579 – 595, 2001.

[3] M. Ries, P. Svoboda, and M. Rupp. Empirical study of subjective quality for massive multiplayer games. In *Systems, Signals and Image Processing, 2008. IWSSIP 2008. 15th International Conference on*, pages 181 –184, june 2008.

[4] P. Svoboda, W. Karner, and M. Rupp. Traffic analysis and modeling for world of warcraft. In *Communications, 2007. ICC '07. IEEE International Conference on*, pages 1612 –1617, june 2007.

[5] Yong Liu, Yang Guo, and Chao Liang. A survey on peer-to-peer video streaming systems. *Peer-to-Peer Networking and Applications*, 1(1):18–28, March 2008.

[6] Philipp Eittenberger, Udo R. Krieger, and Natalia M. Markovich. Measurement and analysis of live-streamed p2ptv traffic. In T. Czachórski, editor, *Performance Modelling and Evaluation of Heterogeneous Networks in HET-NETs 2010*, pages 195–212. IITiS, January 2010.

[7] Thomas Silverston, Olivier Fourmaux, Kavé Salamatian, and Kenjiro Cho. Measuring p2p-tv systems on both sides of the world. In *ICME*, pages 1321–1326, 2010.

[8] Rade Stanojević and Robert Shorten. Trading link utilization for queueing delays: An adaptive approach. *Comput. Commun.*, 33(9):1108–1121, June 2010.

[9] Karthik Lakshminarayanan, Venkata N. Padmanabhan, and Jitendra Padhye. Bandwidth estimation in broadband access networks. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, IMC '04, pages 314–321, New York, NY, USA, 2004. ACM.

[10] A Lakshmikantha, R Srikant, and C Beck. Impact of file arrivals and departures on buffer sizing in core

routers. *IEEE INFOCOM The 27th Conference on Computer Communications*, 19:86–90, 2008.

[11] Amogh Dhamdhere and Constantine Dovrolis. Open issues in router buffer sizing. *SIGCOMM Comput. Commun. Rev.*, 36:87–92, January 2006.

[12] José Saldana, Jenifer Murillo, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete, and José I. Aznar. Evaluation of multiplexing and buffer policies influence on voip conversation quality. *In Proc. CCNC 2011- 3rd IEEE International Workshop on Digital Entertainment, Networked Virtual Environments, and Creative Technology*, pages 1147–1151, Las Vegas, January 2011.

[13] José Saldana, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro, and Luis Casadesus. Influence of online games traffic multiplexing and router buffer on subjective quality. *in Proc. CCNC 2012- 4th IEEE International Workshop on Digital Entertainment, Networked Virtual Environments, and Creative Technology (DENVECT)*, pages 482–486, Las Vegas, January 2012.

[14] Thomas Silverston and Olivier Fourmaux. Measuring p2p iptv systems. In *in Proc. of NOSSDAVâĂŹ07, International Workshop on Network and Operating Systems Support for Digital Audio & Video*, 2007.

[15] Siyu Tang, Yue Lu, Javier Martín Hernández, Fernando A. Kuipers, and Piet Van Mieghem. Topology dynamics in a p2ptv network. In Luigi Fratta, Henning Schulzrinne, Yutaka Takahashi, and Otto Spaniol, editors, *Networking*, volume 5550 of *Lecture Notes in Computer Science*, pages 326–337. Springer, 2009.

[16] B. Fallica, Yue Lu, F. Kuipers, R. Kooij, and P. Van Mieghem. On the quality of experience of sopcast. In *Next Generation Mobile Applications, Services and Technologies, 2008. NGMAST'08. The Second International Conference on*, pages 501–506, September 2008.

[17] Jose Ma Saldana, Julian Fernandez-Navajas, Jose Ruiz-Mas, Eduardo Viruete Navarro, and Luis Casadesus. The utility of characterizing packet loss as a function of packet size in commercial routers. In *CCNC*, pages 346–347, 2012.

[18] Network aware p2p tv application over wise network.

[19] CAIDA. Cooperative association for internet data analysis: Nasa ames internet exchange packet length distributions. Technical report, Cooperative Association for Internet Data Analysis (CAIDA), October 1996.

[20] A Vishwanath, V Sivaraman, and G N Rouskas. Considerations for sizing buffers in optical packet switched networks. *IEEE INFOCOM 2009 The 28th Conference on Computer Communications*, pages 1323–1331, 2009.

[21] Curtis Villamizar and Cheng Song. High performance tcp in ansnet. *SIGCOMM Comput. Commun. Rev.*, 24:45–60, October 1994.

[22] Guido Appenzeller, Isaac Keslassy, and Nick McKeown. Sizing router buffers. *SIGCOMM Comput. Commun. Rev.*, 34:281–292, August 2004.

[23] Neda Beheshti, Yashar Ganjali, Monia Ghobadi, Nick McKeown, and Geoff Salmon. Experimental study of router buffer sizing. pages 197–210, 2008.

[24] Mihaela Enachescu, Yashar Ganjali, Ashish Goel, Nick McKeown, and Tim Roughgarden. Part iii: routers with very small buffers. *SIGCOMM Comput. Commun. Rev.*, 35:83–90, July 2005.

[25] Joel Sommers, Paul Barford, Albert Greenberg, and Walter Willinger. An sla perspective on the router buffer sizing problem. *SIGMETRICS Perform. Eval. Rev.*, 35:40–51, March 2008.

[26] L. A. Casadesus Pazos, J. Fernández Navajas, J. Ruiz Mas, J. M. Saldana Medina, J. I. Aznar Baranda, and E. Viruete Navarro. Herramienta para automatización de medidas de tiempo real extremo a extremo. *Actas del XXVI Simposium Nacional de la Unión Científica Internacional de Radio (URSI 2011)*, Leganés (España). ISBN 9788493393458. Sept. 2011.