Analysis of Capacity Estimation Methods for Real-Time Applications in Internet Accesses

Eduardo A. Viruete[1], Julián Fernández-Navajas[1], Elena Macián-Senz[1], Ignacio Martínez[1], Rafael del-Hoyo[2], José García[1]

[1] Communication Technologies Group (GTC) – Aragon Institute of Engineering Research (I3A)
L.2.05 – Dpt. IEC. Ada Byron Building. CPS Univ. Zaragoza – 50018 Zaragoza (Spain)

[2] Technological Institute of Aragon (ITA) – María de Luna 5 – 50018 Zaragoza (Spain)

[1] {eviruete, navajas, emacian, imr, jogarmo}@unizar.es, [2] rdelhoyo@ita.es

***Abstract*- IP networks and multimedia real-time applications running on them have experienced an amazing development, but require plenty of network resources for a correct performance. The Evaluation of Quality of Service in Internet accesses for Multimedia applications (EQoSIM) system provides an easy way of measuring network performance. Its capacity estimation module needs a special processing to avoid calculation errors due to clock granularity. This paper presents different methods to deal with clock granularity and shows experimental results using a commercial ADSL Internet access.**

I. Introduction

The large increase in the number of users and new multimedia services over Internet generate a significant amount of network traffic. Moreover, the expectation of future growth for the use of multimedia applications such as, e.g. Voice over Internet Protocol (VoIP), indicates that this tendency will increase. In parallel, the heterogeneous characteristics of the different Internet accesses, together with user demands, make it necessary to define the Quality of Service (QoS) [1] that they offer, especially when the accesses provide support to real-time applications.

During the last years, several tools to estimate QoS-related parameters have been developed including bandwidth (BW), delay or packet loss rate (PLR). However, BW is by far the most common parameter used by end users to quantify the performance of their Internet access. The popular Internet connection speed tests [2] make an estimation of BW by measuring the download time of one or more files of a fixed size from different geographically disperse servers. This BW estimation method is fast and simple, but presents several drawbacks, like the fact of being oriented to the estimation of the BW used for file transfers, usually over Transmission Control Protocol (TCP). On the other hand, real-time multimedia applications are usually based on the Real-time Transport Protocol (RTP), which is transported over the User Datagram Protocol (UDP). As a consequence of the different behaviour of TCP and UDP, the current Internet connection speed tests are not so useful for real-time multimedia applications. Thus, our on-line system for the evaluation of QoS (denoted as EQoSIM) [3] is especially focused on real-time multimedia applications and estimates QoS from the end-user point of view, in an automatic way. EQoSIM is able to estimate the most influent QoS parameters in the performance of real-time applications: maximum capacity (C), Available BW (ABW), delay and its variation (jitter), and PLR.

Capacity estimation is a key point in importance. There exist many methods for its estimation, but most of them do not take into account clock granularity (G). This value can be defined as the real time interval in which the system clock measures the same instant, and depends on the operating system and the programming language used. This does not affect if the time measurement equipment is optimized and provided with high precision. Otherwise, a specific analysis is required, as it is presented in this paper, for the problem of how to treat G for a correct estimation of the capacity.

II. QoS Estimation Method and Tests

The capacity estimation method selected for EQoSIM tests been active, one-way, and UDP-based. EQoSIM uses the Packet Pair/Train Dispersion method (PPTD, which considers the end-to-end C) [4] and a modification of the Self-Loading of Periodic Streams (SLoPS) method [3]. The PPTD method is based on sending a burst of k consecutive packets ($k \geqslant 2$) of constant size (S) from the source to the destination. Packet interspacing measured in the destination (temporary separation between packets), allows estimating the maximum rate that can be reached in the network crossed. Therefore, C is estimated using:

$$C = \frac{(k-1)\cdot S}{t_k - t_1} \quad \text{with} \quad \begin{array}{l} t_k : \text{Arrival time of packet } k \\ t_1 : \text{Arrival time of packet } 1 \end{array} \tag{1}$$

However, if the burst is sent when there is cross-traffic at the same time, a sub-estimation of C may happen. This is due to the fact that packets of other traffics mix with the packets of the estimation bursts, increasing the interspacing time measured. Moreover, if k increases, the probability of the additional traffic mixing with the burst increases. Thus, the estimation decreases.

The tests that carry out the estimations have been selected to be used in the highest number of accesses. With regard to the frame length, 4 packet bursts are sent every minute with packets of UDP data lengths 100B, 400B, 700B and 1000B each one, and with an interval of 10s between them. Regarding the number of frames per burst, k, a value of $k = 20$ is enough.

In order to carry out the mathematical operations to obtain the estimation of C, there are different alternatives:

- Not to take into account clock granularity: C is calculated with (1), using directly the timestamps of captured packets.

This work has been supported by projects TSI2004-04940-C02-01, TSI2007-65219-C02-01, and FIT-350300-2007-56 from Science and Technology Ministry (MCYT) and European Regional Development Fund (ERDF), Pulsers II IP IST-27142 from VI Framework, and FPU grant AP-2004-3568 from State Secretary of Universities and Research. The authors would like to express their gratitude to Laura Bueso for her technical collaboration in this article.

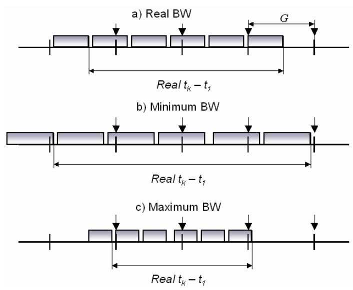


Figure 1. Different arrival possibilities of a burst with 6 frames (*k*=6) in n=4 time blocks.

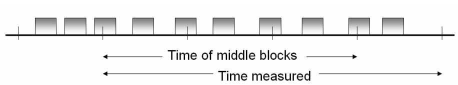


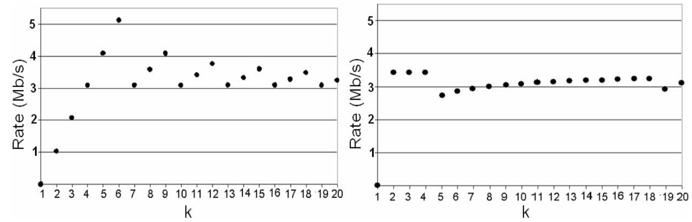
Figure 2. Time used when discarding the first and last blocks.

- Taking into account clock granularity, we obtain the result for C as the range of values maximum-minimum [3], between which the estimated value of C would be (Fig. 1).
- Using the average value of the previous range as the estimation.
- Obtaining the maximum-minimum range for different values of *k* and using the common interval of all of them. In this case, the range maximum-minimum varies as a function of *k*. If in a series of *k* we take the minimum value of the maximum ones and the maximum value of the minimum ones, we obtain a common interval of smaller amplitude.
- Obtaining C from the frames captured, but discarding the frames in the first and last blocks of granularity, and taking the time occupied by the rest of the blocks. This is done because the first and last blocks can be not completely full and could introduce an error (Fig. 2).

III. Results and Discussion

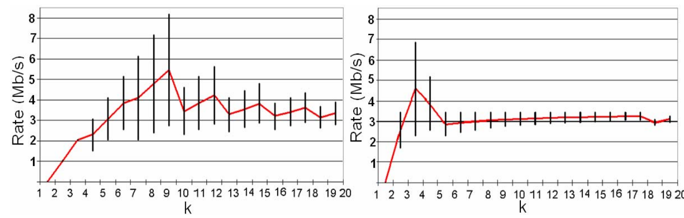
The results obtained by using EQoSIM to determine C in a commercial ADSL Internet access are presented next. 1440 tests were sent during 24 hours over an ADSL access (4Mb/s-DL and 384kb/s-UL), combining traffic in both directions and packet sizes of 100B, 400B, 700B and 1000B.

In the Downlink (DL), the results of C estimations with *S*=100 and *S*=400 are shown in Figs. 3 and 4 (results with *S*=700 and *S*=1000 are not shown because they are very similar to the *S*=400 case). The results presented illustrate the different variants of C estimation taking and not taking G into account. The terminals used for the test have G=10 ms. Fig. 3 shows the results without taking G into account. In this case, the measurements are grouped into blocks, with points forming straight lines of different slopes due to G. On the other hand, Fig. 4 shows the max-min range and the average of possible C values taking G into account. It can be seen that this interval of possible values gets narrower (more precise C estimation) as *k* and *S* increase. The common interval method (not taking the first values of *k* into account) and the method based on discarding the frames in the first and last blocks require high values of *k* and obtain similar results. However, the latter is simpler and requires less processing. In both cases, for the four values of *S*, the C estimation is between 3 and 3.5Mb/s over 85% of the time.



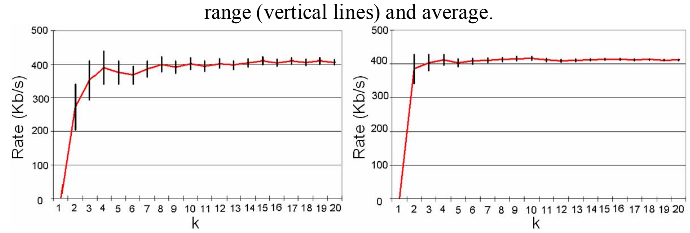
a) *S*=100 b) *S*=400

Figure 3. Capacity estimation in ADSL DL, not taking G into account.



a) *S*=100 b) *S*=400

Figure 4. Capacity estimation in ADSL DL, taking G into account: max-min range (vertical lines) and average.



a) *S*=100 b) *S*=400

Figure 5. Capacity estimation in ADSL UL, taking G into account: max-min range (vertical lines) and average.

In the Uplink (UL), the capacity estimation shown in Fig. 6, for every value of *S* studied, is between 400 and 500kb/s in over 90% of the situations. Similar discussions as those carried out for the DL are valid in this case. The main difference is that C is now much lower (G does not have a great influence because fewer packets fit in a block and the burst occupies more).

It is important to note that all the methods offer similar results when *k* and *S* are large enough compared to G. However, if *k* and *S* are small compared to G, the methods that do not take G into account offer a unique estimation value for a given *k* and *S*, but this value can have a large error. On the other hand, the max-min range method offers a range of values where the estimation lie, which minimum value can be used as the minimum C guaranteed in end-to-end path. The average method also offers a unique value, but this value is closer to the estimation taken as the reference than those values obtained using the methods that do not take G into account. Another method, the common interval method permits to diminish the amplitude of the range of values where the estimation lies. Finally, the method based on discarding the frames in the first and last granularity blocks offers a unique value closer to the reference than the value offered by the average method, but it is not usable for small values of *k*.

References

[1] A. Vogel, B. Kerhervé, G. von Bochmann and J. Gecsei, "Distributed Multimeda and QoS: A Survey", *IEEE Multimedia*, pp. 10 – 18, 1995.

[2] "Bandwidth Speed Test", http://www.bandwidthplace .com/speedtest. Last access: 1/4/07.

[3] E.A. Viruete *et al*., "Evaluation of QoS in Internet accesses for Multimedia applications", *IEEE Consumer Communications and Networking Conference*, pp.356-360, vol. 1, 2006.

[4] C. Dovrolis, P. Ramanathan, D. Moore, "Packet-Dispersion Techniques and a Capacity-Estimation Methodology", *IEEE/ACM Trans. Networking*, vol.12, nº6, pp 963-977, 2004.