Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Networks 50 (2006) 332–349

Computer Networks

www.elsevier.com/locate/comnet

Techniques for available bandwidth measurement in IP networks: A performance comparison

Leopoldo Angrisani [a], Salvatore D'Antonio [b,*], Marcello Esposito [b], Michele Vadursi [a]

[a] *Dip. di Ingegneria Elettrica, Università di Napoli Federico II, Via Claudio 21, 80125 Napoli, Italy*
[b] *CINI—Consorzio Interuniversitario Nazionale per l'Informatica, Via Diocleziano 328, 80124 Napoli, Italy*

Received 1 December 2004; received in revised form 19 May 2005; accepted 6 June 2005
Available online 1 August 2005

Responsible Editor: M. Crovella

Abstract

As the Internet grows in scale and complexity, the need for accurate traffic measurement increases. Among the different parameters relevant to traffic measurement, the paper pays attention to the available bandwidth of a path. In particular, a performance comparison of three different techniques, devoted to available bandwidth measurement, is attained under different operating conditions. The comparison is based on the outcomes of an extensive experimental activity. Experimental tests are not limited to the mere execution of the software tools that implement the techniques under test; indeed, a proper measurement station comprising a digital counter has been set up by the authors with the aim of gaining a reference value to be compared with results provided by the considered tools. The adoption of a performance evaluation methodology relying on the use of electronic instrumentation for time measurement represents a good example of cross-fertilization between two distinct research areas: networking on one side and electronic measurements on the other. The tools have been tested under different cross-traffic conditions and their performance has been evaluated in terms of the following metrics: concurrence, repeatability, bias, and time. For each cross-traffic scenario and with reference to every performance metric, the paper identifies the tool that provides the best results. Furthermore, an optimal setting for the parameters of each tool has been identified thanks to the extensive experimental activity that has been performed.

Keywords: Available bandwidth; Communication network test and measurement; Performance evaluation; Cross-traffic

* Corresponding author. Tel.: +39 81 7682894; fax: +39 81 7682950.
E-mail addresses: angrisan@unina.it (L. Angrisani), salvatore.dantonio@napoli.consorzio-cini.it (S. D'Antonio), marcello.esposito@napoli.consorzio-cini.it (M. Esposito), mvadursi@unina.it (M. Vadursi).

1389-1286/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2005.06.003

1. Introduction

Traffic monitoring provides network operators with a detailed view of the state of their networks. In particular, congestions can be detected through periodic summaries of traffic load and packet loss on individual links; parts of the network exhibiting high delay or loss, as well as routing anomalies such as forwarding loops, can be identified by means of active probes between pairs of nodes in the network. Network operators can exploit results provided by the traffic measurement activity in order to perform fundamental tasks: (i) usage-based accounting, (ii) traffic profiling, (iii) traffic engineering and (iv) attack/intrusion detection.

Among the different metrics peculiar to traffic monitoring, one of the most significant is available bandwidth. As well known, a single link is characterized by its capacity, C, defined as the maximum rate at which it can transfer data. Available bandwidth is related to the unused capacity of a link. More precisely, since at a given instant of time t a link is either transmitting at a rate equal to its capacity C or idle, the definition of instantaneous available bandwidth is meaningless, and a certain time-averaging is needed. The available bandwidth of a network path, made of N links, $L_1, \ldots, L_N$, is, in fact, defined as

$$A(t, \Delta t) = \min_{i=1,\ldots,N} \frac{1}{\Delta t} \int_t^{t+\Delta t} C_i(1 - u_i(\tau))\,\mathrm{d}\tau, \qquad (1)$$

where $C_1, \ldots, C_N$, are the link capacities and $u_1(t), \ldots, u_N(t)$ are the percentages of link utilization.

Estimating the available bandwidth of a network path represents an important task for many applications whose performance and effectiveness can be increased by properly exploiting the output of such activity. For example, available bandwidth evaluation plays a fundamental role in traffic engineering algorithms. For such algorithms, the available bandwidth is one of the major parameters which determine the choice of the best connection path [1].

Different techniques and related software tools, aimed at measuring available bandwidth, have been presented in the literature [2–6]. This paper takes into consideration three of them: (i) Pathload [2,3], which sends packet streams along the path, and tunes the stream rate after analyzing the trend of one-way delays (OWDs), until both an upper and a lower bound for the available bandwidth are found; (ii) IGI [4], which sends packet trains, characterized by increasing gaps between two adjacent packets and then evaluates the available bandwidth on the basis of the variation of the gaps at destination; (iii) pathChirp [6], which launches a number of exponentially spaced probing packet trains, and then performs a statistical analysis at receiver side to determine the available bandwidth.

The authors of the above-mentioned tools, when presenting their work, have provided some experimental results with the aim of evaluating tool performance and demonstrating their effectiveness. Furthermore, several comparative studies of tool performance have recently appeared in the literature. In particular, Strauss et al. [7] have extensively tested Pathload and IGI over 400 different paths, but performance comparison lacks a reliable reference. More precisely, the nominal value of the available bandwidth is derived from Multi-Router Traffic Grapher (MRTG) [8], which is presented as the most accurate method to verify the output of available bandwidth estimation tools. MRTG, however, intrinsically suffers from low resolution.

A comparative evaluation of system capability effects on tool performance has been presented in [9], and available bandwidth estimation techniques have been classified in [10], where some misconceptions and misinterpretations have been highlighted.

Rigorous characterization of all the above-mentioned techniques has not been executed yet. To meet such requirement, the performance of these techniques is here evaluated in terms of measurement repeatability, time and bias with respect to a proper reference value, which is determined through the use of an electronic counter. Performance comparison is then accomplished by putting side-by-side achieved results and evaluating their concurrence, as well. The measurement tools are tested under different cross-traffic conditions, thus extending partial results illustrated in [11], in which only CBR (Constant Bit Rate) background traffic is considered.

The paper is organized as follows: in Section 2 an overview of the tools under study is presented. Section 3 describes the methodology as well as the testbed used to compare the performance of the considered tools. In Section 4 the tools are tested under different network conditions in terms of both link capacity and competing traffic properties. Finally, Section 5 provides some concluding remarks, together with directions of future work.

2. Techniques for available bandwidth measurement: a brief overview

As stated before, three techniques for available bandwidth measurement have been taken into account: Pathload v.1.1.0 [2,3], IGI v.1.0 [4], and pathChirp v.2.3.3 [6].

Pathload consists of two components: the process called SND, which runs at the sender, and the process called RCV, which runs at the receiver. The process SND transmits a periodic UDP (User Datagram Protocol) packet stream to RCV. Let us suppose that the transmission rate of the stream is R bits per second. Since SND timestamps each packet prior to its transmission with a timestamp t_i, RCV can compute the OWD of each packet. Upon the reception of the entire stream, RCV inspects the sequence of OWDs in order to check if the transmission rate R is larger than the available bandwidth A. Indeed, when $R > A$, the OWDs of the stream packets are expected to have an increasing trend. On the contrary, when $R < A$, the OWDs have a non-increasing trend. To actually estimate the available bandwidth along the path, SND and RCV have to cooperate so that R converges iteratively to A. Such measurement methodology is called Self-Loading Periodic Streams (SloPS).

IGI is based on the following technique. A train of probing packets is sent in quick succession and, at the destination, the variation of the packet pair gaps is calculated. Since probing packets travel through the network, cross-traffic packets may be inserted between them, thus increasing the gaps. As a result, the gap value at the destination would be a function of the cross-traffic rate, which can therefore be estimated. To assure that the probing train actually interleaves with cross-traffic, a sequence of packet trains with increasing gaps is sent from source to destination and the difference between the average source and destination gaps is monitored. When such difference becomes practically zero, i.e. cross-traffic is interleaving with probing packets without causing an increase of the gaps, the so-called turning point is reached and the sending process is stopped. IGI finally computes the available bandwidth by subtracting the estimated amount of cross traffic from the bottleneck link capacity.

Based on the concept of "self-induced congestion", pathChirp estimates the available bandwidth along a network path by launching a number of packet chirps, i.e. exponentially spaced probing packet trains, from sender to receiver, and then performing a statistical analysis at the receiver side. More precisely, by measuring packet interarrival times, pathChirp makes an estimate of the per-packet available bandwidth as the instantaneous chirp rate at the packet for which the queuing delay begins increasing. It then takes a weighted average of the per-packet estimates in order to evaluate the per-chirp available bandwidth. By further averaging the per-chirp estimates, pathChirp computes the available bandwidth along the network path.

3. Methodology and experimental setup

3.1. Performance comparison methodology

In order to carry out the experiments, two couples of hosts have been utilized and a suitable network has been set up. For each configuration about one hundred measurements have been performed, so that a statistically significant sample is available; measurement results have then been processed to evaluate their mean value (μ) and experimental standard deviation (σ).

Measurement results have been compared in terms of measurement (i) concurrence, (ii) repeatability, (iii) bias and (iv) time.

With regard to concurrence analysis, measurement results have to be expressed in terms of an interval and related statistical distribution. While IGI and pathChirp provide a single value as the result of a measurement, Pathload gives the available bandwidth variation range. In this light, concerning IGI and pathChirp, the hypothesis of a Gaussian distribution of measurement results is assumed, and the interval considered for concurrence analysis is centred at the mean value and it is six times σ wide. As for Pathload, on the contrary, since no confidence level is associated to Pathload's range, lower and upper extremes of output intervals are averaged, and the measurement result is assumed to be uniformly distributed within the averaged interval. It is worth stressing that measurement results concur if the related intervals overlap.

Repeatability is strictly related to experimental standard deviation. The smaller the experimental standard deviation, the higher the degree of repeatability.

Measurement bias is an indicator commonly used to characterize a measurement system, in general. It gives information about non-zero difference between measured and expected value. In the paper, it has been defined as the difference, Δ, between measurement result and reference value. When investigating measurement bias, the choice of a reference value is crucial; this is particularly true when performance comparison of different measurement solutions is drawn. A proper measurement station, described in the next section, has been set up with the aim of providing a suitable common reference.

The time needed by a measurement tool to perform a single measurement and provide the related result is also taken into account as an indicator of the tool performance. Besides being efficient with regard to time saving issues, tools exhibiting reduced measurement time also allow a better tracking in time of the dynamic behaviour of the measured parameters. Comparison among measurement times of different tools is therefore particularly significant, also in consideration of the fact that available bandwidth is defined as a time-averaged metric.

3.2. Measurement testbed

Fig. 1 shows the testbed set up by the authors. It comprises a sender host and a receiver host, belonging to different LANs and connected

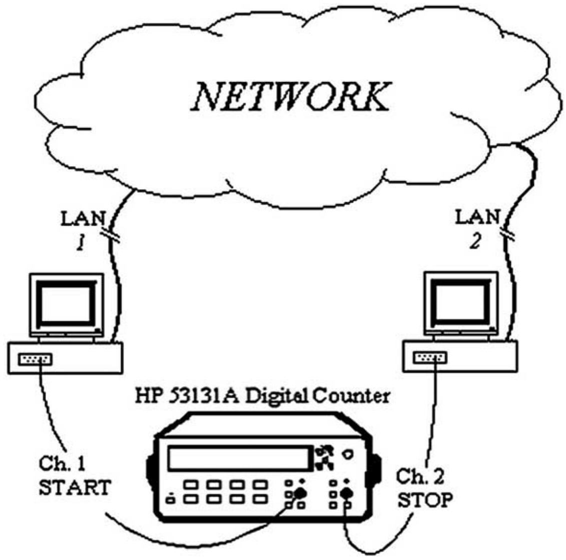

Fig. 1. The measurement station.

through a network of generic topology, on which one or more cross-traffic sources are active. The core of the testbed is a digital counter [12]. Its two channels are connected to, respectively, the sender and the receiver. In particular, channel 1 is connected to pin 7 (RTS, Request to Send) of the serial port of the sender, and channel 2 is connected to pin 4 (DTR, Data Terminal Ready) of the serial port of the receiver.

The idea is to use a traffic generator to transfer data from sender to receiver and then employ the digital counter, rather than uncalibrated host clocks, to perform accurate time interval measurements, from which the reference value of available bandwidth is derived. A suitable traffic generator, named MGEN [13], is adopted to generate UDP cross-traffic. MGEN allows tuning some traffic parameters, such as statistical traffic characterization (i.e. periodic, Poisson), packet size and traffic rate. Another traffic generator, D-ITG [14], is used to generate TCP and HTTP-like traffic. The Distributed Internet Traffic Generator (D-ITG) is a tool capable of producing network traffic and accurately replicating appropriate stochastic processes for both IDT (Inter Departure Time) and PS (Packet Size) independent identically distributed (i.i.d.) random variables. The currently upported protocols are: TCP, UDP, ICMP, DNS, Telnet, VoIP (G.711, G.723, G.729, Voice Activity Detection, Compressed RTP).

The reference value is obtained by calculating the ratio of the amount of transferred data, which is derived from the generator log report, to the transmission time, measured through the electronic counter. Transmission time is measured as the time interval between the rising edges of two voltage pulses, which occur, respectively, on channel 1 and channel 2 of the counter. When the sender starts the transmission, a voltage pulse is generated on its serial port. Then, when the receiver gets the message that the socket has been closed at sender side, a voltage pulse is generated on the serial port of the receiver. The two rising edges of the voltage pulses represent, respectively, the start and the stop event for time interval measurement. In order to generate the two voltage pulses at the right instants, appropriate changes have been made in the source code of the traffic generator.

For each network configuration and cross-traffic amount, the mean value over one hundred measurements is chosen as the reference value. This measure represents the common reference needed for bias evaluation.

3.3. Network under test

Fig. 2 shows the adopted network. Measurement tools under test run on the first couple of hosts (named *Leox* and *Eufrate*), which are equipped with Red Hat Linux (kernel version 2.4.14) and have the same hardware configuration. The other two hosts are used as source and destination for the cross-traffic. All the measurements have been carried out twice, inverting the roles of sender and receiver, in order to detect possible differences between the utilized hosts.

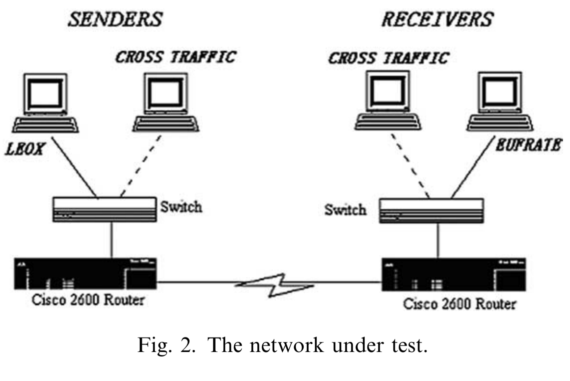

Fig. 2. The network under test.

For each couple of hosts, both the sender and the receiver are connected to a Cisco 3500 Series switch. The two senders belong to the same LAN, and so are the two receivers; therefore, the two senders actually compete to get the required bandwidth. Depending on the network under test, each switch is connected to a Cisco 2600 Series router equipped with two Fast-Ethernet interfaces or a Cisco 3600 Series router equipped with a Fast-Ethernet interface and a 2 Mbps Serial interface.

The choice of such a simple network, comprising only two routers, is motivated by the need to totally control the network under test, in order to apply the proposed method for the reference value determination. In other words, sender and receiver have to be physically close enough, in order to be both connected to the digital counter.

4. Performance comparison

Different network conditions in terms of link capacity and cross-traffic characteristics have been considered. As for the background traffic, the following scenarios have been taken into account: (i) constant bit rate UDP streams, (ii) on/off bursty UDP streams, (iii) multiple TCP connections, (iv) superposition of on/off bursty UDP streams and a TCP connection, and (v) UDP streams with several combinations of Pareto interarrivals and constant or Pareto packet sizes. The considered cross-traffic scenarios represent one of the largest experimental sets presented in the literature. For each scenario and for each considered tool, about one hundred measurements distanced of about 10 s from one another have been made with the aim of granting their independence from one another.

This section is organized as follows. In the first subsection a performance comparison of the employed tools on a network with 10 Mbps capacity is carried out, with respect to all the considered cross-traffic scenarios. In the second subsection differences observed on a slower (2 Mbps) and a faster (100 Mbps) link are illustrated, and possible dependency of the results on network capacity is investigated.

4.1. Network with 10 Mbps bandwidth

It is worth stressing out that each tool has some parameters peculiar to the measurement technique it implements. With respect to tool performance, the most significant parameters are number and size of probing packets for IGI, number and length of streams for Pathload, and measurement time and probing packet size for pathChirp. Authors of the related works have proposed some default values for these parameters, but the experimental evidence suggests that their values might be differently tuned, depending on the cross-traffic nature, in order to achieve better performance. Therefore, for each considered cross-traffic scenario, a prior optimization of the parameters has been experimentally executed and the results obtained with the optimal combination of parameters have then been reported for performance comparison.

4.1.1. UDP cross-traffic with constant bit rate

The first set of experiments has been carried out by using cross-traffic with constant bit rate (CBR). In particular, the percentage of link utilization, u, has been set equal to (i) 20%, (ii) 50% and (iii) 80%. MGEN has been used as cross-traffic generator for this set of experiments.

In these first experiments, default parameter values have proved to be suitable and have therefore been adopted for the comparison. In particular, with regard to IGI, 60 packets of 500 bytes have been used; concerning Pathload, the fleet consists of 12 streams of 100 packets each; as for pathChirp, finally, probing packets of 1000 bytes have been used, the spread factor has been chosen equal to 1.2, and the measurement time has been set equal to 40 s (this is the only parameter that has been modified with respect to its default value of 600 s, which has proved to be too high for the scenario under test). Table 1, which reports related measurement results, is divided into three sections, each corresponding to a different percentage of link utilization. For each section, together with the nominal value, which is only theoretical, a reference value of available bandwidth, achieved through the application of the method proposed in section 3.2, is reported. Results are expressed in terms of

Table 1
Measurement results in the presence of CBR cross-traffic, for different values of u, percentage of usage of the link

	EUFRATE → LEOX			LEOX → EUFRATE		
	IGI	pathChirp	Pathload	IGI	pathChirp	Pathload
Competing traffic = 20% capacity						
	Reference value = 7.566 Mbps			Reference value = 7.465 Mbps		
μ [Mbps]	6.950	8.10	5.60–9.59	6.780	8.07	5.53–9.59
σ%	0.49	6.5	15	0.79	6.5	15
Δ%	8.1	7.1	0.46	9.2	8.1	1.3
Competing traffic = 50% capacity						
	Reference value = 5.2033 Mbps			Reference value = 4.998 Mbps		
μ [Mbps]	4.77	5.66	4.85–9.61	4.88	5.49	4.82–9.63
σ%	6.3	6.4	19	6.2	6.4	19
Δ%	8.3	8.8		2.4	9.8	
Competing traffic = 80% capacity						
	Reference value = 1.613 Mbps			Reference value = 1.623 Mbps		
μ [Mbps]	1.520	1.76	1.07–2.25	1.500	1.74	1.17–2.40
σ%	6.0	8.5	20	6.0	8.5	19
Δ%	5.8	9.1		7.6	7.2	

mean value, μ, experimental standard deviation, σ, and difference Δ between the mean value and the reference value. Both experimental standard deviation and difference between mean and reference value are expressed in percentage relative terms.

With regard to Pathload, difference Δ from the reference value is referred to the centre of the interval, μ, and experimental standard deviation is equal to the amplitude of the interval, scaled by $2\sqrt{3}$.

Fig. 3 aligns and compares the measurement intervals of the different techniques for concurrence analysis purpose. The figure refers to the connection from Leox to Eufrate; the opposite path has, in fact, offered similar outcomes. In particular, horizontal bars represent the intervals that indicate measurement results; three groups of three bars can be singled out, according to the different cross-traffic conditions. In particular, starting from the top of the figure, u is equal, respectively, to 80% for the first, 50% for the second, and 20% for the last set of three bars.

From Table 1 and Fig. 3, the following considerations can be drawn:

- Measurement results provided by IGI are less biased for high cross-traffic rates than for low ones and are characterized by an appreciable degree of repeatability (the relative experimental standard deviation is much inferior to 10%). The value of σ, which is inferior to 1% for the lowest values of u, is noticeable.
- pathChirp gives similar results as IGI in terms of Δ%, while σ% is slightly higher.
- The way Pathload expresses measurement results (an interval) does not allow a clear comparison with other considered techniques based on repeated measurements. Measurement results are, in fact, expressed in terms of an interval which is definitely too wide, as revealed by a large relative standard deviation (up to 20% of the mean value). Even though the reference value falls within that interval, it is difficult to assess whether the tool gives the expected results or not. For the same reason, some cells in the table are intentionally left blank: the choice of calculating Δ as the difference between the reference value and the centre of the interval can lead to high values for Δ% even when the reference value falls within that interval.
- Intervals related to different techniques overlap, and measurements results therefore concur.

4.1.2. On/off bursty UDP cross-traffic

A bursty cross-traffic, i.e. a traffic characterized by a variable bit rate (VBR), has been considered in a large set of experiments. Cross-traffic in real networks is, in fact, typically bursty, rather than CBR, and depending on its burstiness, the available bandwidth can vary dynamically during a measurement period. A characterization of available bandwidth measurement tools should

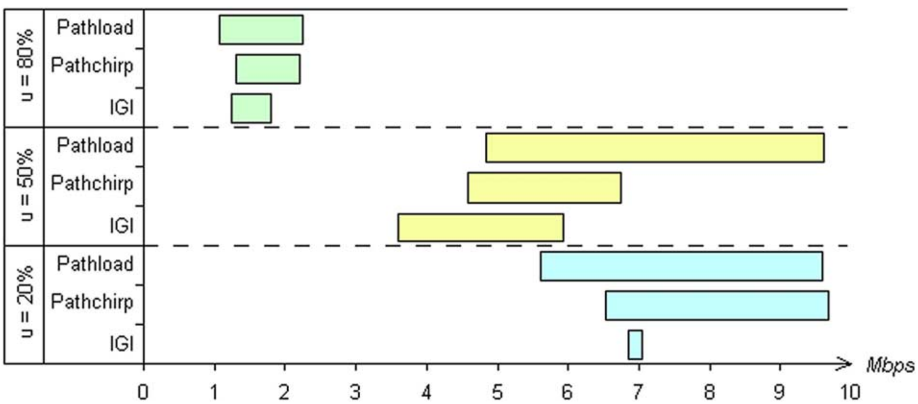

Fig. 3. Concurrence analysis of measurement results in the presence of CBR traffic for different values of link utilization percentage u.

therefore include performance assessment in the presence of bursty cross-traffic.

In particular, UDP streams of on/off bursty traffic have been introduced in our experiments. A source produces on/off bursty traffic when it can operate in only two conditions: in the "on" state, it transmits data at a peak and constant rate, and in the "off" state it does not transmit. In our experiments, the overall source activity, i.e. the alternation of "on" and "off" intervals is periodic, and the percentage of time the source is in the "on" state is referred to as burst duty factor, *df*. The source burstiness, defined as the ratio of the peak to the average transmission rate, can be considered as an indicator of the source activity: the higher the burstiness, the more different the source behaviour from that of a CBR source (CBR sources are characterized by a unitary burstiness). Following this definition, it is easy to realize that source burstiness is the reciprocal of *df*.

Experiments with UDP streams characterized by different burst duty factors (25%, 50% and 75%), different values of peak transmission rate, *p* (20%, 40%, and 80% of the link capacity), and a period equal to 2 s, have been carried out. In this set of experiments, MGEN has been used as traffic generator.

For the sake of brevity, numerical results related to only some representative cross-traffic scenarios are given. From such results, the dependence of tool performance on parameters values can be inferred.

Before comparing the tool outputs, details regarding the tuning of the tool parameters, with regard to their default values, are given.

IGI: Concerning IGI, experiments have been made with 60 (default value), 120 and 240 probing packets; packet sizes of 500 (default value), 750, 1000, 1250 and 1400 bytes have been chosen. Larger packet sizes, bigger than the MTU (Maximum Transmission Unit), would imply the fragmentation of the packet into two smaller packets and, consequently, have been ignored.

For the sake of clarity, Table 2 shows measurement results achieved in the presence of cross-traffic characterized by $df = 0.25$ and $p = 8$ Mbps. Results related to all the combinations of the tool parameter values taken into account are reported in the table and are expressed in terms of μ, $\sigma\%$ and $\Delta\%$.

Experimental results highlight that an increase in the number of probing packets does not generally provide better results in terms of $\Delta\%$, difference between measured and reference values. In the few cases where a slight reduction of $\Delta\%$ (a few tenths percentage points) is experienced, significant increases in both measurement time and tool intrusiveness counterbalance it.

On the contrary, increasing packet size has turned out to be a good way of enhancing the tool performance. More precisely, the choice of smaller probing packets has resulted in an underestimate of available bandwidth equal to about 6–12% of the reference value, while for packet sizes of

Table 2
Measurement results provided by IGI, in the presence of on/off bursty UDP cross-traffic characterized by $df = 0.25$ and $p = 8$ Mbps, for different values of the tool parameters, number of packets *N* and packet size

	Packet size					
	500 bytes	750 bytes	1000 bytes	1250 bytes	1400 bytes	
μ [Mbps]	7.61	8.07	8.41	8.71	8.75	$N = 60$ packets
$\sigma\%$	8.6	5.8	4.4	2.1	5.4	
$\Delta\%$	12	6.4	2.4	1.0	1.5	
μ [Mbps]	7.66	8.11	8.47	8.72	8.74	$N = 120$ packets
$\sigma\%$	6.3	4.6	2.7	1.9	4.6	
$\Delta\%$	11	5.9	3.8	1.2	1.4	
μ [*Mbps*]	7.74	8.13	8.52	8.78	8.82	$N = 240$ packets
$\sigma\%$	5.2	3.9	2.9	1.7	5.1	
$\Delta\%$	10	5.7	1.2	1.9	2.3	

The reference value, measured through the counter, is 8.62 Mbps.

1000, 1250 and 1400 a slight overestimate, always within 2% of the reference value, has been experienced. The optimal choice turns out to be 60 packets of 1250 bytes.

Two reasons may be adduced to explain such erroneous estimate in case of small packet sizes, as already highlighted in [4], and both of them depend on the fact that the gap value at the turning point proportionally grows with the packet size. Such behaviour is shown in Fig. 4, which plots the difference between initial and output gap as a function of the initial gap, for probing packets consisting, respectively, of (a) 500, (b) 750, (c) 1000, and (d) 1250 bytes.

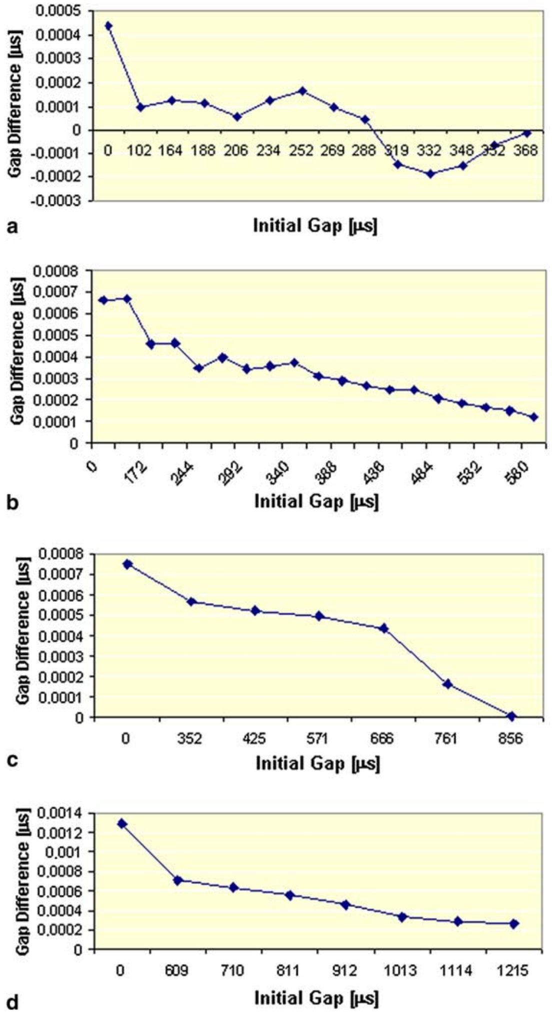

Fig. 4. Values of initial gap and difference between initial and output gap for single measurement using IGI, with probing packets consisting of (a) 500, (b) 750, (c) 1000, and (d) 1250 bytes.

In detail, the fact that smaller packets result in smaller gaps has two effects: (i) the resulting probing train is more sensitive to the burstiness of the cross-traffic, and therefore the convergence of the gap difference to zero is oscillating and less regular than the one experienced with larger probing packets; as a consequence, higher values of experimental standard deviation and reduced measurement repeatability are experienced. Moreover, (ii) it is harder to correctly generate and measure smaller gap values.

As a further example, Table 3 shows measurement results achieved in the presence of cross-traffic characterized by $df = 0.5$ and $p = 4$ Mbps. The average cross traffic rate is 2 Mbps, the same as in the previous example, while the burstiness is halved. The results confirm that Δ is influenced by packet size, rather than number of packets. The best choice of parameters, i.e. the one that grants the minimum Δ%, is 120 packets of 1400 bytes.

pathChirp: The first noticeable issue regarding parameter values for pathChirp is that default measurement time is set equal to 600 s, while our experience for the network under test has shown that much shorter times (20–40 s) are enough to assure good measures. As for packet size, which is the other parameter taken into account, we used packets of at least 1000 bytes, following the authors' suggestion [6], and not larger than the MTU, in order to avoid fragmentation.

Table 4 depicts the results in the presence of a cross-traffic characterized by values of df and p equal to 0.25 and 8 Mbps, respectively. The results show that the tool overestimates the available bandwidth for longer measurement times and provides more accurate measurement results for larger packets. The optimal measurement time, i.e. the one granting the lowest Δ% (inferior to 2% for packets of 1200 and 1400 bytes), has come out to be equal to 20 s.

Pathload: Let us consider again the scenario where the cross-traffic is characterized by $df =$

Table 3
Measurement results provided by IGI, in the presence of on/off bursty UDP cross-traffic characterized by $df = 0.5$ and $p = 4$ Mbps, for different values of the tool parameters, number of packets N and packet size

	Packet size					
	500 bytes	750 bytes	1000 bytes	1250 bytes	1400 bytes	
μ [Mbps]	7.34	7.61	7.77	8.12	8.29	$N = 60$ packets
σ%	8.9	7.5	5.6	4.1	5.3	
Δ%	14	11	8.9	4.8	2.8	
μ [Mbps]	7.40	7.65	7.84	8.24	8.34	$N = 120$ packets
σ%	8.2	6.9	4.8	3.8	4.7	
Δ%	13	10	8	3.4	2.2	
μ [Mbps]	7.48	7.74	7.94	8.35	8.37	$N = 240$ packets
σ%	7.9	6.4	4.4	3.4	4.5	
Δ%	12	9.3	6.9	2.1	1.9	

The reference value, measured through the counter, is 8.53 Mbps.

Table 4
Measurement results provided by pathChirp, in the presence of on/off bursty UDP cross-traffic characterized by $df = 0.25$ and $p = 8$ Mbps, for different values of the tool parameters, packet size and measurement time

	Pkt size = 1000 bytes Meas. time = 20 s	Pkt size = 1200 bytes Meas. time = 20 s	Pkt size = 1400 bytes Meas. time = 20 s	Pkt size = 1000 bytes Meas. time = 40 s	Pkt size = 1000 bytes Meas. Time = 60 s
μ [Mbps]	8.38	8.55	8.76	8.91	9.11
σ%	5.1	3.4	2.3	4.6	4.2
Δ%	2.8	0.81	1.6	3.4	5.7

The reference value, measured through the counter, is 8.62 Mbps.

0.25 and $p = 8$ Mbps. With the default values of $N = 12$ streams and $L = 100$ packets, Pathload significantly underestimates available bandwidth (the actual available bandwidth is about four times the measured one). The tool therefore seems to be totally inaccurate in the presence of a bursty traffic characterized by a low *df*. As the experiments have proved, however, such inaccuracy is only due to the inadequateness of the default values, and accurate measurements can be achieved by properly fixing them. As shown in Table 5, which gives the results of sets of measurements carried out with different values of N and keeping L equal to 100 packets, reducing the number of streams does not provide significant improvements, the estimated value being still very distant from the actual one. On the contrary, a reduction of the number of probing packets in each stream is the correct way of tuning the parameters and granting good performance. Specifically, Table 5 also reports the results achieved by successively modifying the value of L, while keeping N equal to its default

Table 5
Measurement results provided by Pathload, in the presence of on/off bursty UDP cross-traffic characterized by $df = 0.25$ and $p = 8$ Mbps, for different values of the tool parameters, number of streams, N, and stream length, L

	$N = 12$ $L = 100$	$N = 8$ $L = 100$	$N = 6$ $L = 100$	$N = 4$ $L = 100$	$N = 2$ $L = 100$	$N = 1$ $L = 100$	$N = 12$ $L = 50$	$N = 12$ $L = 40$
μ [Mbps]	1.475	1.56	2.025	2.095	2.92	6.05	8.915	7.38
σ%	6.0	7.0	14	15	10	10	1.5	4.3

The reference value, measured through the counter, is 8.62 Mbps.

value of 12 streams; the optimal value for L is 50 packets. The reason for this behaviour of the tool essentially lies in its iterative operation: with default parameter values almost every stream is labelled as being characterized by an increasing OWD trend, i.e. each stream is affected by cross-traffic, even though the latter is present only for 25% of measurement time and, consequently, the estimate converges to a value that is definitely lower than the expected one. By reducing the stream length, there are more streams not affected by cross-traffic and not labelled as being characterized by an increasing OWD trend; therefore, the bursty nature of cross-traffic is better tracked down by the tool. On the other hand, performance tends to worsen when streams consisting of less than 50 packets are used; we conjecture that this worsening is probably due to the limited number of samples considered to evaluate the OWD trend. In conclusion, 50 packets seem to be the optimal choice in the presence of bursty traffic.

Only after tuning the parameters of all the tools, an intelligible and significant performance comparison can be gained. To this aim, Table 6 shows measurement results provided by the tools, in different scenarios of bursty cross traffic; in particular, different duty factors have been considered, for a given $p = 8$ Mbps. From the analysis of the results, the following considerations can be drawn:

- Experimental standard deviation, σ, is generally lower for lower values of df.
- At the same time, in the presence of a low bursty duty factor ($df = 0.25$), difference between measured and reference value, Δ, is significantly lower than for higher values of df. The tools exhibit good performance, in terms of measurement bias, especially for $df = 0.25$ ($\Delta = 1.0\%$ for IGI and $\Delta = 0.8\%$ for pathChirp).
- Measurement bias results provided by Pathload are definitely worse than those provided by IGI and pathChirp. Indeed, except for $df = 0.25$, $\Delta\%$ is bigger than 13% and $\sigma\%$ equal to 48%, whereas $df = 0.75$ accounts for meaningless results.
- The best results in terms of measurement bias are granted by pathChirp, while IGI results exhibit a slightly lower experimental standard deviation. Such considerations are generally valid, regardless of the value of df.
- Measurement results provided by the three tools generally concur.

Some further considerations on measurement time can be drawn:

Table 6
Measurement results provided by all the considered tools, in the presence of on/off bursty UDP cross traffic characterized by $p = 8$ Mbps and different values of df

	IGI	pathChirp	Pathload
$df = 0.25$			
Reference value = 8.62 Mbps			
μ [Mbps]	8.71	8.55	8.67–9.16
$\sigma\%$	2.1	3.4	2.1
$\Delta\%$	1.0	0.80	3.4
$df = 0.5$			
Reference value = 6.75 Mbps			
μ [Mbps]	7.25	7.1	3.82–7.86
$\sigma\%$	3.6	5.6	20
$\Delta\%$	7.4	5.2	13
$df = 0.75$			
Reference value = 4.43 Mbps			
μ [Mbps]	4.85	4.12	0.64–6.85
$\sigma\%$	5.2	6.2	48
$\Delta\%$	9.5	6.9	15

- Measurement time is an input parameter for pathChirp. In the large number of experiments carried out, 20 s are enough to achieve a good bandwidth estimate. A measurement time greater than 20 s does not generally provide better results in terms of both lower Δ and σ.
- IGI converges more slowly when more packets and/or larger packets are utilized. With regard to experiments carried out with $df = 0.25$, for instance, when 60 packets of 500 bytes are used, measurement time is about 20 s; if 60 packets of 1000 bytes are used, measurement time is about 30 s; measurement time grows up to respectively about 40 s and 60 s when packet size is raised to 1400 bytes and the number of packets is increased to 240. Similarly, the higher the cross-traffic rate, the higher the measurement time.
- With regard to Pathload, when default parameter values are adopted, measurement time is generally longer than 60 s, and becomes longer than 80 s for higher cross-traffic rates. Even if the stream length is halved, measurement time is not shorter than 50 s.
- In conclusion, pathChirp proves to be definitely more effective in terms of reduced measurement time, whereas Pathload is characterized by the highest measurement time.

4.1.3. Multiple TCP connections

Most of Internet applications adopt TCP as transport protocol. Scenarios where cross-traffic consists of TCP connections, as well as a superposition of bursty UDP streams and TCP connections, are therefore realistic, and it is worth including them in our set of experiments.

In particular, experiments involving the combination of multiple periodic TCP streams have shown that tool performance does not vary significantly as a function of the number of streams and their average rate. Consequently, for the sake of brevity, Table 7 shows measurement results provided by all the considered tools under the respective optimal parameter choices, when cross-traffic consists of two 2 Mbps TCP periodic streams. The table also reports the optimal parameter combinations. The three tools provide concurrent results (the related intervals are not disjoint) and experimental standard deviation is within few percents. As in the on/off bursty UDP cross-traffic scenario, pathChirp grants a reduced measurement bias, whereas IGI accounts for a lower experimental standard deviation.

4.1.4. Superposition of on/off bursty UDP streams and a TCP connection

Further experiments have been carried out considering combinations of bursty UDP traffic and TCP connections. Indeed, a mixed UDP–TCP traffic is a quite common situation in the Internet scenario. Table 8 accounts for the measurement results provided by the tools in the presence of cross-traffic consisting of a combination of a bursty 8 Mbps UDP stream, with $df = 0.25$, and one or two 1 Mbps TCP streams, respectively. IGI and pathChirp give better results in terms of measurement bias (Δ between 1% and 4%), whereas the experimental standard deviation characterizing Pathload results is lower than the others (4–6% against 6–8%). On the overall, the three tools provide good results both in terms of Δ% and σ%.

Table 7
Measurement results provided by all the considered tools, under the respective optimal parameter choices, in the presence of cross-traffic consisting of two 2 Mbps TCP periodic streams

	IGI	pathChirp	Pathload
	60 packets Packet size = 1000 bytes	Meas. time = 20 s Packet size = 1200 bytes	40 streams Stream length = 150 packets
Reference value = 6.04 Mbps			
μ [Mbps]	5.78	5.85	5.89–6.67
σ%	5.3	6.4	7.2
Δ%	4.3	3.2	3.9

Table 8
Measurement results provided by the tools in the presence of cross-traffic consisting of a combination of a bursty 8 Mbps UDP stream, with $df = 0.25$ and, respectively, one or two 1 Mbps TCP streams

	IGI	pathChirp	Pathload
	60 packets Packet size = 1250 bytes	Meas. time = 20 s Packet size = 1200 bytes	12 streams Stream length = 50 packets
Reference value = 7.58 Mbps			
μ [Mbps]	7.49	7.42	7.49–8.62
σ%	8.1	6.3	8.1
Δ%	1.2	2.1	6.3
Reference value = 6.43 Mbps			
μ [Mbps]	6.67	6.28	6.68–6.96
σ%	7.5	8.2	2.4
Δ%	3.7	2.3	6.1

4.1.5. UDP streams with combinations of Pareto packet interarrivals, and constant or Pareto packet sizes

Due to the Internet traffic complexity and variability, it is difficult to gain a unique model for simulating and analyzing its behaviour [15]. The evolution of networks and applications has required and still requires a continuous tuning of traffic models [16–21]. A number of studies, based on the analysis of aggregated traffic, have proved that traffic on the Internet has a self-similar or fractal nature [16,22]. Research activities have been also conducted on the effects of traffic characteristics on network performance [23, 24].

Some of the results of those studies have inspired the experiments that are described in the following. In the first set of experiments, in particular, self-similar traffic has been considered [19], packet interarrivals being modelled as i.i.d. random variables of a Pareto distribution, with a mean value of 3 ms and shape factor $\alpha = 1.5$. Generated packets consist of 1470 bytes. For the sake of brevity, the outcomes of three configurations are shown, in which cross-traffic consists respectively of one and three UDP streams with the aforementioned statistical characteristics. Related results are reported in Table 9; both Δ% and σ% are slightly higher than in the other experiments, but they still represent acceptable values, Δ% remaining within some percents.

Some experiments where the cross-traffic is made up of packets characterized by both interarrivals and sizes with a Pareto distribution have also been carried out, according to the results presented in [16]. In particular, packet interarrivals have the same statistical characterization of the previous experiments, while packet sizes are i.i.d. Pareto random variables, with a mean equal to 2000 bytes and a shape factor $\alpha = 1.21$. As shown in Table 10, in such cases IGI provides worse results than the other two tools ($\Delta = 17\%$ and $\sigma = 12\%$). The reason might be found in the high variability that has been imposed on cross-traffic, which could prevent IGI from correctly reaching the turning point.

When adopting Pareto distribution, strong statistic dependence between packet interarrivals is imposed. A recent paper [25] has, indeed, put in evidence that the long-range dependence (LRD) of packet interarrivals is an invariant characteristic for time scales longer than 1 s, whereas for time scales up to 100 ms, the statistical correlation is much weaker. Since the algorithms for available bandwidth estimation, on which the three considered tools are based, mainly generate packets at time intervals inferior to 100 ms, further experiments have been carried out by adopting a different traffic model. More precisely, cross-traffic consists of packets whose sizes are distributed as Pareto i.i.d. random variables (mean $L = 4100$ bytes, $\alpha = 1.35$) and whose interarrival times are distributed as exponential i.i.d. random variables. Different values of parameter λ, which is the reciprocal of the mean interarrival time, have been taken into account for the experiments ($\lambda = 64$,

Table 9
Measurement results provided by the tools in the presence of UDP cross-traffic characterized by interarrival times modeled as i.i.d. random variables of a Pareto distribution, with a mean value of 3 ms and shape factor $\alpha = 1.5$

	IGI	pathChirp	Pathload
	60 packets Packet size = 1250 bytes	Meas. time = 30 s Packet size = 1200 bytes	12 streams Stream length = 50 packets
1 UDP stream			
Reference value = 7.01 Mbps			
μ [Mbps]	7.49	7.38	7.15–7.83
σ%	8.7	5.9	5.2
Δ%	6.8	5.3	6.8
	60 packets Packet size = 1250 bytes	Meas. time = 20 s Packet size = 1200 bytes	12 streams Stream length = 50 packets
3 UDP streams			
Reference value = 6.18 Mbps			
μ [Mbps]	6.43	5.78	5.18–6.58
σ%	8.8	6.7	14
Δ%	0.39	6.5	4.9

Generated packets consist of 1470 bytes.

Table 10
Measurement results provided by the tools in the presence of UDP cross-traffic characterized by interarrival times modeled as i.i.d. random variables of a Pareto distribution, with a mean value of 3 ms and shape factor $\alpha = 1.5$, and packet sizes modeled as i.i.d. Pareto random variables, with a mean equal to 2000 bytes and a shape $\alpha = 1.21$

	IGI	pathChirp	Pathload
	60 packets Packet size = 1250 bytes	Meas. time = 30 s Packet size = 1200 bytes	12 streams Stream length = 50 packets
Reference value = 1.79 Mbps			
μ [Mbps]	2.1	1.67	1.75–1.98
σ%	12	8.7	7.1
Δ%	17	6.7	4.2

160, 256 packets/s); expected percentage of link utilization u has been evaluated according to $u \cdot C = \lambda \cdot L$ ($u = 20\%$, 50%, 80%). All the related results are given in Table 11. While IGI suffers from values of both Δ% and σ% even higher than 10% and up to 19%, values of Δ% and σ% experienced with pathChirp are always inferior to 10% (less than 3% when $u = 20\%$).

4.2. Further scenarios

As already stated, experiments carried out on 10 Mbps capacity link have been repeated on a slower link, without changing network topology. In particular, the two routers have been connected through a 2 Mbps link. The first noticeable issue on such slower link concerns Pathload, which either does not provide results, or provides too wide intervals. Performance comparison in this case has therefore been limited to IGI and pathChirp.

In a constant bit rate traffic scenario, IGI exhibits higher values of Δ% on the 2 Mbps link than on the 10 Mbps one. Its performance is worse for higher values of u, reaching peaks of more than 15% when u is equal to 80%. Irrespectively of u, pathChirp generally overestimates available bandwidth, and its results are characterized by higher experimental standard deviation, σ being generally greater than 10%.

Table 11
Measurement results provided by the tools in the presence of cross-traffic characterized by interarrival times distributed as exponential i.i.d. random variables, and packet sizes modeled as i.i.d. Pareto random variables, with a mean equal to 4100 bytes and a shape $\alpha = 1.35$

	IGI	pathChirp	Pathload
u = 20%			
Reference value = 7.688 Mbps			
μ [Mbps]	8.34	8.12	7.65–8.43
σ%	7.4	2.8	5.6
Δ%	8.5	5.6	4.6
u = 50%			
Reference value = 5.027 Mbps			
μ [Mbps]	5.68	5.22	4.97–5.75
σ%	11	8.4	8.4
Δ%	13	3.8	5.9
u = 80%			
Reference value = 2.281 Mbps			
μ [Mbps]	2.72	2.09	2.18–2.83
σ%	14	9.0	15
Δ%	19	8.3	9.9

When bursty UDP traffic is considered, IGI does not seem to benefit from a probing packet size increase, contrary to expectations and experimental evidence on the 10 Mbps link; values of Δ% are generally worse with respect to the previous network configuration. Concerning pathChirp, experimental standard deviation is definitely too high ($\sigma > 20\%$), and available bandwidth is overestimated. If measurement time is increased, however, lower values of Δ% and σ% are experienced, both being within 7%.

In the presence of cross-traffic consisting of a superposition of TCP streams, a good repeatability is experienced with both tools, along with little difference between measured and reference available bandwidth values. Similar outcomes are observed when a UDP stream is also present, except for values of Δ% related to pathChirp, which overestimates available bandwidth even more than 10% over reference value.

Finally, contrary to the 10 Mbps scenario, IGI provides satisfying results in the presence of cross-traffic consisting of UDP streams with combinations of Pareto packet interarrivals, and constant or Pareto packet sizes, whereas pathChirp does not confirm the performance shown on the 10 Mbps link, irrespective of measurement time value.

The last set of experiments has been performed on a network with 100 Mbps capacity. It has to be noted that the three tools give better results in the CBR background traffic scenario. More precisely, IGI shows optimal performance in case of $u = 20\%$ and $u = 50\%$, Δ% being smaller than 4%. This behaviour contrasts with performance observed in the 10 Mbps scenario, where IGI provides less biased results for higher cross-traffic rates. When $u = 80\%$, the best results are provided by Pathload, which, anyway, exhibits a satisfying behaviour even when the link utilization is lower. With regard to pathChirp, its performance improves as the percentage of competing traffic grows. Experimental standard deviations related to the three tools are comparable to those experienced in the 10 Mbps scenario.

In the presence of UDP bursty traffic, IGI performance does not particularly benefit from an increase in either the number of probing packets, or their size. All experienced values of both Δ% and σ% are, in fact, higher than 7%. As already experienced in the 10 Mbps scenario, the best estimates attained by using pathChirp refer to a measurement time equal to 20 s, which is much lower than default value. Experimental standard deviation, however, is higher than 10%. Results provided by the three tools with bursty UDP cross-traffic concur.

With respect to the 10 Mbps scenario, results with TCP cross-traffic are less satisfying. Specifically, both $\Delta\%$ and $\sigma\%$ are notably higher, for all the considered tools. The most representative example regards pathChirp, which suffers from bandwidth underestimation, and exhibits values of $\Delta\%$ generally greater than 10–15%.

Finally, in the presence of UDP traffic with Pareto packet interarrivals, IGI turns out to be the most performing tool, even if it overestimates the actual available bandwidth, while pathChirp does not confirm the good performance shown in the 10 Mbps scenario, thus replicating the behaviour observed in the 2 Mbps network. In particular, pathChirp results are characterized by a $\sigma\%$ greater than 20%.

5. Conclusions

The concept of bandwidth is central to packet-switched networks since it refers to the amount of data a link or a network path can transfer per unit of time. Several applications can benefit from knowing the bandwidth characteristics of network paths traversed by their traffic. In fact, bandwidth estimation allows applications such as file transfers or multimedia streaming to optimize their performance. The bandwidth available to an application directly impacts its performance. Several techniques have been defined to measure the available bandwidth of a link or a path, but a rigorous characterization and performance comparison is missing.

This paper has aimed at comparing the performance of three different tools that employ diverse strategies to estimate the available bandwidth. The considered tools are: Pathload, which measures the available bandwidth by sending packet streams and analyzing the trend of one-way delays at the receiver side; IGI, which evaluates the available bandwidth on the basis of the variation of the gaps between two adjacent packets at destination; pathChirp, which estimates the available bandwidth by sending exponentially spaced probing packet trains and measuring packet interarrivals. Comparison of results has not been limited to mean values, but it has also been based on concurrence, repeatability and bias of the measurements. With the aim of evaluating the latter, a reference value has been obtained by means of a suitable measurement station set up by the authors. Moreover, with regard to possible values of several parameters, such as cross-traffic type, cross-traffic rate and direction of the probing traffic, a large set of experimental tests has been carried out.

What has clearly come out from the experiments is a good repeatability for all three techniques in the 10 Mbps scenario, as proved by experimental standard deviations of few percentage units. This has been true for both IGI and pathChirp, whereas in some cases $\sigma\%$ experienced with Pathload has been higher than 15%, due to the way it outputs measurement results. Experimental standard deviation has become higher as cross-traffic variability has increased. This phenomenon has been more evident on faster (100 Mbps) and slower (2 Mbps) links. At the same time, results have shown that it is impossible to determine which technique and related software implementation are absolutely the best, since the performance has varied depending on cross-traffic characteristics, and all the techniques have presented a certain bias.

In case of cross-traffic with constant bit rate, intervals representing measurement results of the three tools have concurred. PathChirp has given similar results as IGI in terms of $\Delta\%$ in all the considered scenarios, whereas its $\sigma\%$ has been generally higher. Pathload results have generally been characterized by high values of both $\Delta\%$ and $\sigma\%$, since the output of the tool is an interval rather than a single value. In the 100 Mbps scenario, however, the best results with high link utilization have been attained by using Pathload.

In the scenario characterized by bursty cross-traffic, as well as when cross-traffic has consisted of multiple TCP connections, results provided by the three tools have concurred.

More precisely, in case of on/off bursty UDP cross-traffic, pathChirp has shown the most robust behaviour with respect to the variations of cross-traffic parameters in the 10 Mbps scenario, whereas IGI has performed better than the other tools in terms of both $\Delta\%$ and $\sigma\%$ in both the 100 Mbps and 2 Mbps scenarios. Finally, Pathload has proved to be inadequate to this scenario.

In the presence of competing traffic consisting of two TCP periodic streams characterized by a bit rate equal to 20% of the link capacity, IGI and pathChirp have shown similar overall performance: a better bandwidth estimate is paid with a larger standard deviation. However, pathChirp has exhibited values of $\Delta\%$ generally greater than 10–15% in the 100 Mbps scenario, and it is therefore inadequate. Also with this kind of cross-traffic, Pathload has performed worse than the other two tools.

Cross-traffic statistical distributions have also been considered. It has to be noticed that in the 10 Mbps scenario with competing traffic characterized by constant packet size and Pareto packet interarrivals, the three tools have provided values of both $\Delta\%$ and $\sigma\%$, which have been slightly higher than those observed in the same scenario with different cross-traffic. In the other capacity scenarios, results have generally been worse when competing traffic interarrivals are Pareto distributed.

On the whole, the analysis carried out in this work has shown that the tools under study reach considerable performance in all the capacity scenarios, provided that the cross-traffic presents a low degree of variability. Indeed, the trend has revealed an increase in result degradation as traffic variability grows.

Future research activity will aim at executing further experimental tests on more complex network topologies and developing a new and more powerful technique for available bandwidth measurement, capable of overcoming most flaws identified in the analyzed ones.

Acknowledgment

This work has been partly supported by the European Union under the E-Next Project FP6-506869.

References

[1] M. Kodialam, T.V. Lakshman, Dynamic routing of restorable bandwidth-guaranteed tunnels using aggregated network resource usage information, IEEE/ACM Transactions on Networking 11 (3) (2003) 399–410.

[2] M. Jain, C. Dovrolis, End-to-end available bandwidth: measurement methodology, dynamics and relation with TCP throughput, in: Proceedings of ACM SIGCOMM, August 2002.

[3] M. Jain, C. Dovrolis, Pathload: a measurement tool for end-to-end available bandwidth, in: Proceedings of Passive and Active Measurement Workshop, March 2002.

[4] N. Hu, P. Steenkiste, Evaluation and characterization of available bandwidth probing techniques, IEEE JSAC Special Issue in Internet and WWW Measurement, Mapping, and Modeling 21 (6) (2003) 879–894.

[5] C. Dovrolis, P. Ramanathan, D. Moore, What do packet dispersion techniques measure? Proceedings of IEEE INFOCOM 2 (Apr.) (2001) 905–914.

[6] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, L. Cottrell, pathChirp: efficient available bandwidth estimation for network paths, in: Proceedings of Passive and Active Measurements Workshop, April 2003.

[7] J. Strauss, D. Katabi, F. Kaashoek, A measurement study of available bandwidth estimation tools, in: Proceedings of Internet Measurement Conference, Miami, Florida, October 2003.

[8] T. Oetiker, Monitoring your IT gear: the MRTG story, IEEE IT Professional 3 (6) (2001) 44–48.

[9] G. Jin, B.L. Tierney, System capability effects on algorithms for network bandwidth measurement, in: Proceedings of Internet Measurement Conference, Miami, Florida, October 2003.

[10] M. Jain, C. Dovrolis, Ten fallacies and pitfalls on end-to-end available bandwidth estimation, in: Proceedings of Internet Measurement Conference, Taormina, Italy, October 2004.

[11] L. Angrisani, S. D'Antonio, M. Vadursi, G. Ventre, Performance comparison of different techniques for available bandwidth measurement in packet-switched networks, in: Proceedings of VECIMS'03, Lugano, Switzerland, July 2003, pp. 212–217.

[12] Agilent 53131A/132A/181A Counters. High-performance, low-cost counters simplify and speed systems and bench frequency measurements, Product Overview, Agilent Technologies Literature no. 5967-6039EN, May 2004.

[13] NRL Protocol Engineering Advanced Networking Research Group, The Multi-Generator (MGEN) Version 4.0—User's and—reference Guide. Available from: <http://mgen.pf.itd.nrl.navy.mil/mgen.html>.

[14] S. Avallone, D. Emma, S. Guadagno, A. Pescapè, G. Ventre, D-ITG: distributed internet traffic generator, in: Proceedings of QEST2004, September 27–30, 2004, Enschede (Netherlands).

[15] S. Floyd, V. Paxson, Difficulties in simulating the internet, IEEE/ACM Transactions on Networking 9 (4) (2001) 392–403.

[16] W. Willinger, V. Paxson, M.S. Taqqu, Self-similarity and heavy tails: structural modeling of network traffic, A

Practical Guide to Heavy Tails, Chapman & Hall, New York.
[17] V.A. Bolotin, Y. Levy, D. Liu, Characterizing data connection and messages by mixtures of distributions on logarithmic scale, in: Proceedings of ITC-16, Edinburgh, UK, June 1999, pp. 887–894.
[18] V. Paxson, Empirically-derived analytic models of wide-area tcp connections, IEEE/ACM Transactions on Networking 2 (4) (1994) 316–336.
[19] V. Paxson, S. Floyd, Wide area traffic: the failure of Poisson modeling, IEEE/ACM Transactions on Networking 3 (3) (1995) 226–244.
[20] K.V. Cardoso, J.F. de Rezende, Design and Use of an aggregated HTTP model, in: Proceedings of IEEE International Conference on Communications, April 2002.
[21] J. Cao, W.S. Cleveland, D. Lin, D.X. Sun, On the nonstationarity of internet traffic, in: Proceedings of ACM SIGMETRICS/Performance, Cambridge, MA, June 2001.
[22] M.E. Crovella, A. Bestavros, Self-similarity in world wide web traffic: evidence and possible causes, IEEE/ACM Transactions on Networking 5 (6) (1997) 835–846.
[23] K. Park, G.T. Kim, M.E. Crovella, On the effect of traffic self-similarity on network performance, in: Proceedings of SPIE International Conference on Performance and Control of network Systems, November 1997.
[24] M.E. Crovella, Performance evaluation with heavy tailed distributions, Lecture Notes in Computer Science, vol. 1786, March 2000, pp. 1–9.
[25] Z.-L. Zhang, V.J. Ribeiro, S.B. Moon, C. Diot, Small time scaling behaviors of Internet backbone traffic: an empirical study, in: Proceedings of INFOCOM2003.

Leopoldo Angrisani was born in Nocera Superiore, SA, Italy, on April 16, 1969. He received the M.S. degree (cum laude) in electronic engineering from the University of Salerno, Salerno, Italy, in 1993 and the Ph.D. degree in electrical engineering from the University of Napoli Federico II, Napoli, Italy, in 1997. Since 2002, he has been an Associate Professor with the Dipartimento di Informatica e Sistemistica, University of Napoli Federico II. He is involved in research on new methods based on the wavelet transform for detecting, measuring, and classifying transient signals, new methods based on chirplet and warblet transforms for instantaneous frequency estimation, new methods based on time–frequency transforms for testing RF equipment for mobile communications, new measurement procedures for communication networks test and measurement, and design, realization, and characterization of VXI instruments based on digital signal processors.

Salvatore D'Antonio received the M.S. degree in computer engineering from the University of Naples "Federico II". He is currently a researcher at C.I.N.I., the Italian University Consortium for Computer Science. His current research interests include network monitoring and control, Quality of Service (QoS) provisioning over IP networks and e-business platforms. He is currently involved in a number of international research projects in the area of interdomain QoS monitoring and delivery.

Marcello Esposito received the degree in Telecommunications Engineering from the University of Napoli "Federico II in 2000. He is currently working as a researcher at C.I.N.I., the Italian University Consortium for Computer Science and he teaches Computer Programming at the University of Napoli. His research interests include QoS provisioning, interdomain network management and content adaptation for web services.

Michele Vadursi was born in Naples, Italy, on May 9, 1978. He received the M.S. degree (cum laude) in telecommunication engineering from the University of Naples 'Federico II', Naples, Italy, in 2002. He is currently pursuing the Ph.D. degree in electrical engineering at the same University. His research activity is focused on QoS evaluation in both wired and wireless LAN's, RF transmitter testing and troubleshooting, and instantaneous frequency estimation through digital signal processing.