Bandwidth estimation in networks, an overview

Fabian Jäger*

September 23, 2010

The performance of a network depends on several conditions which are often not exactly measurable but can be approximated. These information are needed by administrators to scale a network or an application which needs to adapt its behavior to the current network conditions. For example, the performance of many network applications like VoIP software or P2P content distribution systems mainly depends on the available bandwidth of the network. Therefore several tools have been developed to estimate the available bandwidth and capacity. These approaches vary in techniques and focus of the measurement. All these tools have in common that their accuracy and efficiency are still limited. This paper reviews, evaluates and discusses several papers approaches to this issue.

Keywords: Bandwidth estimation, network capacity measurement, active probing, passive probing

1 Introduction

Measuring a network is an important task to improve the performance of the network itself, or the applications that use it. Therefore it is necessary to get as detailed information as possible which is not always easy. Often, we have no access to routers, as for example on the Internet, and we are forced to estimate parameters like capacity or bandwidth. This work will focus on the measurement of the maximum available bandwidth and the techniques to determine it. We will give an overview of the tools and techniques and discuss relevant papers dedicated to this issue.

The purpose of this background research is the design of an adaptive, bandwidth-aware streaming strategy for real-time video data in a conferencing system called PlaceCam by Daviko [1]. For real time video streaming it is important to know the available bandwidth to adjust the amount of video packets. The codec has the feature

*E-mail: `fabian.jaeger@haw-hamburg.de`

to drop specific packets after encoding which leads to a reduction of resolution. The advantage is to encode a frame just once for all receivers but still have the ability to reduce the amount of traffic for receivers with less bandwidth. If the amount of data is too high, it will jam the network and the stream will stall. On the other hand, if a lot of frames are dropped, the network can handle the stream, though the quality might be low. The challenge is to find the max. available bandwidth to send a stream with good quality without stalling.

This paper has seven parts. In chapter 2 the terms are defined. Chapter 3 contains the problem description. Chapter 4 describe the basic ideas. Chapter 5 describe the measurement techniques. Chapter 6 gives an overview of the available tools and papers. The conclusion follows in Chapter 7.

2 Definition of terms

First of all we must clarify some terms. The route a packet traverses in a network is called path. This path consists of hops (transitions) which are connected by links. Each link has a capacity which is its max. possible bandwidth. This value is fixed and mainly depends on the hardware in use. The capacity of the path is determined by the link with the smallest capacity.

The traffic we generate for measurement is called probing packets in this paper. The other traffic is called competing traffic or cross traffic. This traffic reduces the available bandwidth and can occur on every link of a path.

Since TCP connections have a special behavior on a network path, extra terms exist to describe them. The reason is the flow-control mechanism of TCP. This will be discussed in chapter 5.5. The capacity of a TCP connection is called Bulk Transfer Capacity (BTC) whereas the max. possible data-transfer-rate is the TCP throughput.

3 Problem description

The aim of this paper is to discuss the possibilities to estimate the available bandwidth of a path. The challenge is to find methods to measure a path without access to the routers. The delay on a path itself consists of three factors.

Serialization delay The serialization delay is the time needed by a packet to traverse the link. The duration depends on the length of the packet and is proportional to packet size/capacity.

Propagation delay The propagation delay is independent of the packet size and describes the time a single bit needs to cross the link.

Queuing delay Queuing delays occur on routers or switches, if competing traffic occupies ports. The device must store the packets as long as the outgoing port is blocked. So the delay heavily depends on the amount of competing data on the path and is independent of the packet size.

The simplest scenario is a network path with no network traffic. If a network path contains no competing traffic, the available bandwidth is the same as the capacity. If competing traffic occurs on one of the links, the queuing delay increases and infects the measuring. This paper discusses techniques to interpret these delays correctly to make an estimation about the capacity and available bandwidth.

The measurement software has two approaches to measure a network-path. It can run on a single machine and use the TTL field of IP like traceroute, or use a client-server architecture, which is used in most cases. If we use the client-server architecture we can only measure the path with access to both machines.

The measurement duration is a very important issue for applications that use this information to scale the network traffic. If the measurement takes too long, the measured data might be too old already. Thus we try to identify techniques where measurement speed and accuracy are balanced.

A Further problem is the amount of network traffic generated by the measurement. If it is too intrusive it can affect the other traffic or even jam the path.

In summary we need a measurement technique, which is fast, accurate and affects the other traffic as little as possible.

4 Basic Ideas

The available bandwidth estimation tools can be categorized in two major techniques.

The **Probe Gap Model** (PGM) sends a pair of packets with a predefined gap between it. At the end of the path the gap is measured again and compared to the initial gap. Every delaying of the second packet, for example if the network is too slow or a packet from the competing traffic arrives between the packet pair, will lead to an increased gap. This is shown in figure 1.

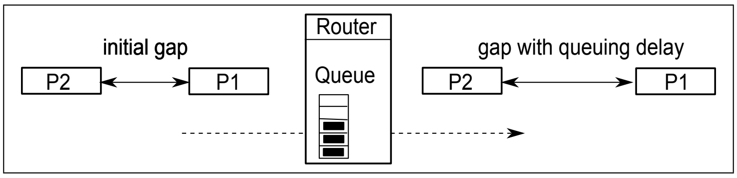

Figure 1: PGM measures the gap at the beginning and at the end of a path

The difference between initial gap and the gap at the end of the path is called dispersion. Every time a queuing delay occurs, the dispersion increases. PGM can be used to measure the capacity and the available bandwidth. If we use it to measure the capacity, we try to reduce the impact of the competing traffic by using a small gap and determine the capacity of the path with the dispersion. If we try to estimate the available bandwidth we use the dispersion to measure the competing traffic on the tight link and subtracts it from the capacity of the bottleneck. The result is the available bandwidth. The problem of this method is the assumption that the tight link occurs at the bottleneck. This might be the case in most of the scenarios but is not always the case.

The **Probe Rate Model** (PRM) uses a self-induced congestion on the path. The rate of the probing packets increases slowly and is measured again at the end of the path. If the rate gets higher then the available bandwidth, the rate is slowed down by the tight link, which is noticeable at the end of the path.

The figure 2 shows the one way delay which is measured by the receiver while the probing rate (R) increases. The one way delay does not change as long as the probing rate is lower than the available bandwidth (A). If the probing rate is higher than the available bandwidth we can measure the increasing one way delay.

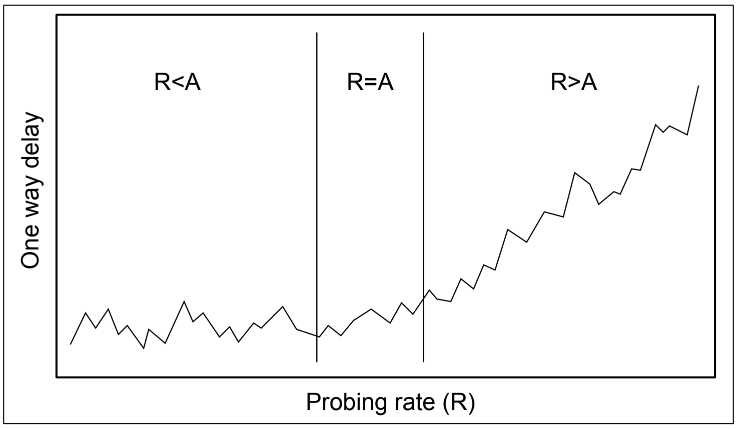

Figure 2: Results from PRM

We try to find the area where the probing rate is equal to the available bandwidth. The PRM takes longer than the PGM and also needs to jam the path, which might slow down the competing traffic.

5 Measurement Techniques

Over the past few years several techniques have been developed to measure the relevant characteristics of a network. Several interesting aspects in a network can be measured. Each of them needs another technique. In addition to that every network has different characteristics which affects the techniques and tools. Here we concentrate on the major techniques. The first two techniques are used to estimate the capacity of a path, which is an important information for the remaining techniques to estimate the available bandwidth.

5.1 Variable Packet Size Probing

The VPS was explored by Bellovin [2] and Jacobson [11] and tries to measure the capacity of each hop along a path. The time-to-live (TTL) field of the IP header is used to let the probing packets expire at a predetermined hop. With the returning ICMP error message it is possible to determine the round-trip-time (RTT) which depends on the serialization delay, propagation delay and the queuing delay.

The time needed for the ICMP replies is included in propagation delay, because the size is independent of the size of the probing packet.

For each packet size, the measurement is repeated several times and finally the minimum measurement is used as RTT. The reason for taking the minimum is given by the assumption that in this case no or just a short queuing delay occured. The minimum measured $RTT(L)$ to a specific hop i is expected to be:

$$RTT_i(L) = \alpha_i + L * \sum_{k=1}^{i} \frac{1}{C_k} \tag{1}$$

where

L is the packet size of the probing packet

C_k is the capacity of the kth hop

α_i is the delay of a single bit to the hop i which does not depend on L.

The delay between two hops is calculated as follows:

$$T_{i,i-1}(L) = RTT_i(L) - RTT_{i-1}(L) \tag{2}$$

After we measured $RTT_i(L)$ and $RTT_{i-1}(L)$ to calculate $T_{i,i-1}(L)$, the capacity C_i between hop $i-1$ and i can be calculated as follows:

$$\begin{aligned}
T_{i,i-1}(L) &= \Big(\alpha_i + L * \sum_{k=1}^{i} \frac{1}{C_k}\Big) - \Big(\alpha_{i-1} + L * \sum_{k=1}^{i-1} \frac{1}{C_k}\Big) && (3)\\
T_{i,i-1}(L) &= \alpha_i - \alpha_{i-1} + L * \Big(\sum_{k=1}^{i} \frac{1}{C_k} - \Big(\sum_{k=1}^{i} \frac{1}{C_k} - \frac{1}{C_i}\Big)\Big) && (4)\\
T_{i,i-1}(L) &= \alpha_i - \alpha_{i-1} + L * \frac{1}{C_i} && (5)\\
C_i &= \frac{L}{T_{i,i-1}(L) - (\alpha_i - \alpha_{i-1})} && (6)
\end{aligned}$$

A problem of the VPS are store-and-forward switches because they are invisible to the measurement but generate packet size dependent delays. This leads to bandwidth underestimation. Also the TTL-expire method can be a problem if some hops do not reply with an ICMP message. A big advantage is that it is not necessary to run a measurement-tool at the other end of the path.

5.2 Packet pair/train dispersion

PPTD is a PGM technique that was explored by Jacobson [10], Keshav [14], and Bolot [3]. It is used to measure the capacity of an end-to-end path. A pair of packets of equal size is sent with a gap. The gap is the time between the last bit of the first packet and the last bit of the second packet. On the other end of the path the gap is measured again and compared to the initial gap. The change of the gap at the end of the path is called dispersion. Only the link with the smallest capacity influences the dispersion. Every link, the packets pass, can increase the dispersion if the capacity is lower than the capacity of the previous link. If the capacity of the previous link is lower than the current the dispersion stays the same. So only the slowest link characterises the dispersion independent of all other links. Until now we have assumed a path without competing traffic. But the result can be inexact caused by capacity underestimation or overestimation if a link is also used by cross traffic. If the cross traffic slows the first packet but not the second, the dispersion decreases and results in overestimation of the link. On the other hand, capacity underestimation occurs if the cross traffic arrives at a switch between the two probing packets. In this case the dispersion is increased because the second packet is slowed down.

Several methods exist to determine the correct capacity because the standard statistical evaluations, for example the median, are often incorrect [5].

Another idea to reduce the impact of the cross traffic is to use more than just two packets. This packet train delivers the same result as the packet pair if no cross traffic jams. Otherwise the packet train is much more insensitive than the packet pair.

5.3 Self-loading periodic streams

SLoPS is a PRM technique to estimate the available end-to-end bandwidth of a path [12]. A number of equal sized packets is sent at a certain rate. If the rate is higher than the available bandwidth, it will be delayed because the packets are queued or dropped at the tight link. Otherwise, if the rate is lower than the available bandwidth, the receiver will get the packets with the same rate they are sent with. SLoPS tries different rates and uses a binary search to approximate the bandwidth. Between each rate a delay is included so that the path is not jamed all the time. This makes the measurement very slow, but is necessary to keep the network useable for other users or applications. Nevertheless the network will be jamed for a short time and may slow down other network traffic. Another problem that may happen is a variation of the competing network traffic since the measurement takes a long time. In this case, SloPS notices this variation and returns a rate window with the range of bandwidth.

It is not necessarily needed but we can use the estimated capacity from VPS or PPTD to find a good start rate for SLoPS. For example a 100 Mbit/s path with 10 Mbit/s competing traffic is tested with SLoPS. In this case we can estimate the 100 Mbit/s to determine 50Mbit/s as start rate for the binary search.

5.4 Trains of packet pairs

TOPP is a PGM technique, that sends a train of packets with a certain gap between the packets to measure the available bandwidth. The basic idea of TOPP is the same as PPTD but this time we try to measure the competing traffic. TOPP uses also a gap between a train of packets and measures the dispersion at the end of the path. Every packet of the competing traffic, which arrives at a router at the same time, may be inserted between the packets of the train and increases the gap. Like in PPTD where only the slowest link characterises the dispersion only the tight link with the least available bandwidth characterises the bandwidth of the competing traffic. For example if we have 3 links with L1: 10 Mbit/s, L2: 100 Mbit/s and L3 1000 Mbit/s and competing traffic on each link with C1: 5 Mbit/s, C2: 90 Mbit/s and C3: 100 Mbit/s the link L1 is the tight link in this scenario and increases the dispersion the most.

To determine the available bandwidth we also need the capacity. In the example above we could measure the capacity of the bottleneck, which is L1 in this case. The available bandwidth would be

$$\begin{aligned} available\ bandwidth &= capacity - competing\ traffic \\ &= 10\ Mbit/s\ -\ 5\ Mbit/s \\ &= 5\ Mbit/s \end{aligned}$$

We assume that the bottleneck and the tight link are the same link, but if it is not the case the available bandwidth cannot be estimated correctly. For example could the competing traffic on L2 be 97 Mbit/s. L2 would be the new tight link, while L1 would still be the bottleneck. This problem and its effect on the measurement is very well discussed in the paper [9]. They also discuss a PGM tool which do not need the capacity for calculating the available bandwidth to avoid this problem.

5.5 TCP Measurement

All techniques discussed so far are operating on OSI layer 3 and do not consider the characteristics of the TCP protocol. Since 90% of the Internet traffic uses TCP, we need special techniques to measure the available bandwidth of a TCP connection [15].

The throughput measurement of TCP differs from the previously discussed approaches because there are many factors which may influence the measurement. First of all the specifications and implementations of TCP vary (NewReno, Reno and Tahoe) which can influence the measurement. Also the slow-start mechanism of TCP influences the throughput of little data much more than the available bandwidth. Good examples are webpages which often consist of text and a few pictures. On the other hand, a ftp connection for example is often used to transfer a big amount of data. In this case the slow-start does not inflict the throughput noticeable. In this case the type of competing traffic (TCP or UDP) and the number of TCP connections inflict the throughput. TCP shares the available bandwidth with other TCP flows fairly in contrast to UDP which does not. So for example if we want to measure the possible

TCP throughput where the competing traffic is also a TCP connection which saturates the link, the before discussed measurement techniques would return 0 Mbit/s available bandwidth as result, because the link is fully saturated. This result is wrong, because TCP would share the connection and the correct available bandwidth is capacity/2. If the competing traffic is UDP there will be no sharing and the available bandwidth may approach zero.

6 Available Applications

Since the measurement tools are unprecise, the development of new tools is still active. As discussed before they mainly differ in the used technique an in what they shall measure. In this chapter the most important tools are listed and classified in the tables 1, 2 and 3.

Table 1: TCP tools

Application	Version	Methodology	Last release
treno[a]	970214	BTC	2000-05-16
cap	-	BTC	
ttcp[b]	2.01.01.11	TCP throughput	2010-02-04
Iperf[c]	2.0.3	TCP throughput	2008-03-18
Netperf[d]	2.4.5	TCP throughput	2009-06-12

[a] http://www.psc.edu/networking/treno_info.html
[b] http://www.pcausa.com/Utilities/pcattcp.htm
[c] http://iperf.sourceforge.net/
[d] http://www.netperf.org/netperf/

Table 2: Capacity tools

Application	Version	Methodology	Last release
pathchar[a]	-	VPS	1998-11-18
clink	-	VPS	
pchar[b]	1.5	VPS	2005-02-12
bprobe	-	PPTD	
nettimer[c]	2.3.8	PPTD	2002-01-03
pathrate[d]	2.4.1	PPTD & TOPP	2006-09-01
sprobe	-	PPTD	

[a] ftp://ftp.ee.lbl.gov/pathchar/
[b] http://www.kitchenlab.org/www/bmah/Software/pchar/
[c] http://www.hpl.hp.com/personal/Kevin_Lai/projects/nettimer/index.html
[d] http://www.cc.gatech.edu/fac/Constantinos.Dovrolis/bw-est/

Table 3: Available bandwidth

Application	Version	Methodology	Last release
pathload[a]	1.3.2	SLoPS	2005-05-19
Abing[b]	2.2.0	TOPP	2004-09-16
IGI[c]	2.1	TOPP	2006-05-25
pathChirp[d]	2.4.1	SLoPS	2005-06-07
ASSOLO[e]	0.9a	SLoPS	2008-11-04
DietTop [f]	0.1	SLoPS	2004-11-22
Yaz[g]	0.1	SLoPS	2005-12
PTR[h]	2.1	SLoPS	2006-05-25
Spruce[i]	0.2	TOPP	2003-12-11

[a] http://www.cc.gatech.edu/fac/Constantinos.Dovrolis/bw-est/
[b] http://www-iepm.slac.stanford.edu/tools/abing/
[c] http://www.cs.cmu.edu/ hnn/igi/
[d] http://www.spin.rice.edu/Software/pathChirp/
[e] http://netlab-mn.unipv.it/assolo/
[f] http://www.idt.mdh.se/ ajn12/index.phtml?choice=software
[g] http://wail.cs.wisc.edu/waildownload.py
[h] http://www.cs.cmu.edu/ hnn/igi/
[i] http://project-iris.net/software.html

6.1 Comparison

A lot of comparisons have been made between this tools. Often just a few tools are compared to each other and the test set-ups differ. So it is not easy to use the results to find the "best" application. Also the tools are developed for different operation areas.

First of all the paper from R. Prasad et al. [16] should be mentioned because it is the base of this writing. A good overview of the measurement techniques and implemented tools is given. All techniques are described and easy to understand. This paper is probably the best entry point to the topic. The description of the tools includes the operation area, the technique it uses and a brief description of how it works and how it differs from related tools. The performance of the tools is not compared and so no statements are made concerning of accuracy, speed of measurement or the amount of probing packets.

In the paper [9] IGI, PTR and pathload are compared to each other. The used technique of IGI described accurately provides a very good insight into the work of SLoPS. It should be mentioned that this paper is hardly understandable for beginners. It does not give any overview of other tools/techniques. Otherwise this paper is very helpful for readers with experience. Many special characteristics of problems that may occur in a network are described. IGI is explained on different theoretical networks with different set-ups of IGI. The theoretical result is verified on real Internet paths. This is different from other papers which uses little self-made networks as test set-up. Because the tools are often used in the Internet which may be different from a small network, this paper might be interesting for internet measurements. Also an interesting phenomenon is considered if the bottleneck is not the tight link. This can happen if on another link a lot of competing traffic occur. This leads to different measurement results.

Base on the results of the measurements some specifications are made how to set-up IGI optimal. The parameter discussed are the initial gap, the size of the probing packets and the length of the train.

The conclusion is that IGI and PTR yield almost the same results with equal accuracy, speed and amount of data while pathload takes longer to estimate the bandwidth but has the same accuracy.

In a similar paper R. Kapoor et al. [13] describe the tool CapProbe and compare it with pathchar and pathrate. The mechanism of CapProbe and the packet pair technique and described and furthermore how much the packet size influences the measurement. A very good analysis is given about the different kinds of competing traffic and how much they interact with the probing traffic. The tools are simulated on a 6 hop network with different kinds of competing traffic. Afterwards the tools are tested and compared on the Internet. The authors came to the conclusion that CapProbe has a similar accuracy as pathrate whereas Pathchar is less accurate. Also CapProbe seems to be better in terms of speed.

A good comparison of the different tools to estimate the available bandwidth is given by paper [7]. The tested tools are Abing, ASSOLO, DietTopp, IGI, pathChirp, Pathload, PTR, Spruce and Yaz. As test set-up a small self-made 100 Mbps network is used with two routers and two switches and it is loaded with 0-64 Mbps competing traffic. The tests show that pathChirp often overestimate the available bandwidth what seems to be a well known problem [3] [4]. IGI and PTR increase their accuracy when the competing traffic increases while Abing, DietTopp and Spruce do the opposite. Also the duration to estimate the available bandwidth and the amount of probing packets is measured and compared with each other. The writers came to the conclusion that ASSOLO, Pathload and Yaz are very accurate but Abing seems to be the best choice if you need a fast measurement. Unfortunately they only used a simple and small network for comparison. It would be interesting to see how the same tests would act on the Internet.

An other considerable paper concerning the TCP throughput is written by M. Jain and C. Dvorolis [12]. They give a detailed description of SLoPS and give an overview of TOPP. They implemented the tool Pathload which uses the SLoPS methodology. They use a 3 hop network for testing and afterwards a few Internet paths. Interesting is the comparison between the measured available bandwidth and the real tcp throughput.

J. Strauss et al. compared IGI, Pathload and Spruce on more than 400 different Internet paths [17]. They came to the conclusion that Spruce is more accurate than Pathload and IGI while it uses a relative low amount of probe traffic. Another very detailed comparison of these three tools and additional Abing and Pathchirp is made by C. Guerrero and M. Labrador [8]. They use a 4 hop network for testing.

A very short comparison between a PGM-based tool (Spruce) and a PRM-based tool (Pathload) is made by G. Urvoy-Keller et al [18]. It presents a good and short overview concerning the work of PGM and PRG and how they behave on a network

7 Conclusion

Overall the measurements are still unstable and return different results on different test set-ups. Often small self-made network to compare different tools with each other. The advantage of the testbed is the ability to check the results. The capacity, the used hardware, the path the packets will take and the competing traffic are all parameters that are known in this case. Also it is possible to inspect every single hop and get informations directly from the routers and switches.

On the other side it is almost impossible to build a testbed which represents the Internet. The paths in the Internet have a lot more hops and links, use different hardware and use other protocols. Also the competing traffic can change on a link. So it is important to test the tools as well on the Internet to see how stable they are outside of a sheltered network. A big problem on the Internet is the unknown structure of a path. We get no information about the hardware, the protocols and the real competing traffic. Furthermore we do not have access to the routers and switches. These problems make it impossible to verify the measurement results. We only can use more tools and assume that the average of the results will be the right value. In a few papers like [9] and [13] an internet measurement has been made.

Another problem is the absence of references or standard test set-ups. So it is hard to compare the different results of the papers, particularly because everybody uses another test set-up. The measurement tools deliver different results on different networks. For example VPS measurement tools return unexact results if the path contains store-and-forward switches (cf. [16] s.30). Another example is described in [9] where the results depends on the position of the bottleneck and tight link.

References

[1] Placecam. http://www.placecam.com.

[2] Steven M. Bellovin. A best-case network performance model. http://www.cs.columbia.edu/~smb/papers/netmeas.pdf, 1992.

[3] Jean-Chrysostome Bolot. Characterizing End-to-End Packet Delay and Loss in the Internet. *Journal of High Speed Networks*, 2:305–323, 1993.

[4] Robert L. Carter and Mark E. Crovella. Measuring bottleneck link speed in packet-switched networks. *Perform. Eval.*, 27-28:297–318, 1996.

[5] Constantinos Dovrolis, Parameswaran Ramanathan, and David Moore. What do packet dispersion techniques measure? In *IN PROCEEDINGS OF IEEE INFOCOM*, pages 905–914, 2001.

[6] Constantinos Dovrolis, Parameswaran Ramanathan, and David Moore. Packet-dispersion techniques and a capacity-estimation methodology. *IEEE/ACM Trans. Netw.*, 12(6):963–977, 2004.

[7] Emanuele Goldoni and Marco Schivi. End-to-end available bandwidth estimation tools, an experimental comparison. In Fabio Ricciato, Marco Mellia, and Ernst W. Biersack, editors, *TMA*, volume 6003 of *Lecture Notes in Computer Science*, pages 171–182. Springer, 2010.

[8] Cesar D. Guerrero and Miguel A. Labrador. On the applicability of available bandwidth estimation techniques and tools. *Comput. Commun.*, 33(1):11–22, 2010.

[9] Ningning Hu and Peter Steenkiste. Evaluation and characterization of available bandwidth probing techniques. *IEEE Journal on Selected Areas in Communications*, 21:879–894, 2003.

[10] V. Jacobson. Congestion avoidance and control. In *SIGCOMM '88: Symposium proceedings on Communications architectures and protocols*, pages 314–329, New York, NY, USA, 1988. ACM.

[11] Van Jacobson. Pathchar: A tool to infer characteristics of internet paths. ftp://ftp.ee.lbl.gov/pathchar/msri-talk.pdf, 1997.

[12] M. Jain and C. Dovrolis. End–to–End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput. In *Proc. ACM SIGCOMM'02*, pages 295–308, New York, NY, USA, 2002. ACM.

[13] Rohit Kapoor, Ling-Jyh Chen, Li Lao, Mario Gerla, and Yahya Sanadidi. Capprobe: a simple and accurate capacity estimation technique. *SIGCOMM Comput. Commun. Rev.*, 34(4):67–78, 2004.

[14] Srinivasan Keshav. A control-theoretic approach to flow control. *SIGCOMM Comput. Commun. Rev.*, 21(4):3–15, 1991.

[15] Sean McCreary and kc claffy. Trends in wide area ip traffic patterns - a view from ames internet exchange. *ITC Specialist Seminar, Monterey, CA*, 2000.

[16] Ravi Prasad, Constantinos Dovrolis, Margaret Murray, and Kimberly C. Claffy. Bandwidth Estimation: Metrics, Measurement Techniques, and Tools. *IEEE Network*, 17(6):27–35, November–December 2003.

[17] Jacob Strauss, Dina Katabi, and Frans Kaashoek. A measurement study of available bandwidth estimation tools. In *IMC '03: Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 39–44, New York, NY, USA, 2003. ACM.

[18] Guillaume Urvoy-Keller, Taoufik En-Najjary, and Alessandro Sorniotti. Operational comparison of available bandwidth estimation tools. *SIGCOMM Comput. Commun. Rev.*, 38(1):39–42, 2008.