Reducing the Effects of Cross Traffic in Packet-Pair Based Bottleneck Capacity Measurements[1]

Nischal M. Piratla Abhijit A. Bare Anura P. Jayasumana Rick Whitner*
CNRL, Department of Electrical and Computer Eng., Colorado State University, Fort Collins, CO.
**Agilent Technologies, 4380 Ziegler Rd., Fort Collins, CO.*

Abstract

The accuracy of the currently available bottleneck capacity measurement tools that use the packet pair technique is significantly affected by cross traffic and queuing delays. The impact of these factors is investigated, and a new technique is presented to overcome these effects by estimating the jitter based on packets sent around the same time as the packet pair. A tool based on this principle, Captool, measures the end-to-end bottleneck capacity by reducing these adverse effects, thereby increasing the accuracy of the end-to-end measurements.

1. Introduction

End-to-end capacity measurements can assess a network's suitability to handle specific applications. These measurements can also be used to troubleshoot and diagnose network problems.

Almost all end-to-end capacity measurement tools [2][3][4][5] use the packet pair technique [1]. According to this technique, two packets are sent along a multi-hop network. When a packet first passes through a narrow bandwidth link, it spreads out in time due to lower transmission speed. We use the term 'spreading' to define this phenomenon. When this spreading occurs in the first packet, the new gap (the dispersion) between the first bits of the first and second packets increases. When the packets reach the destination, the gap created by the slowest link is carried forward. If D is the dispersion for packets with length L, the bottleneck capacity B is equal to L/D. In the presence of cross traffic (CT) and queuing delays in the network, this dispersion value is distorted and the above fails to give accurate results [2][6]. Below, we present a technique to compensate for these effects.

2. Jitter compensation technique

Below, we show that the error in dispersion due to CT and queuing delays is, in fact, equivalent to the jitter value that packets spaced apart undergo in transit. Let d_i denote the delay between packets A and B in the i^{th} hop due to CT and queuing. It includes the effects of all the delays due to CT and queuing until the i^{th} hop, but not the effects due to the packet spreading.

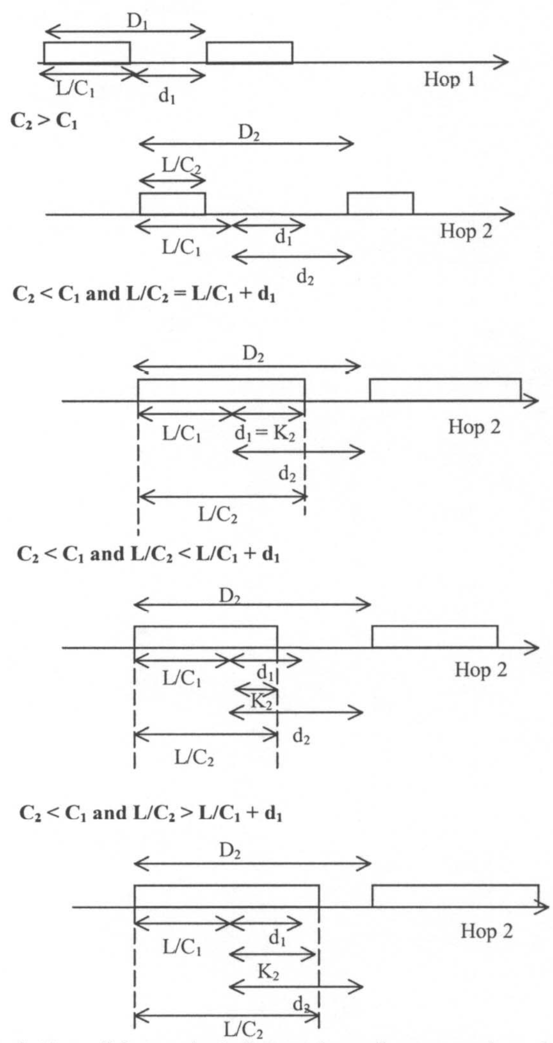

Figure 1. Possible packet dispersions from one hop to the other

[1] This research was supported in part by NSF Information Technology Research Grant No. 0121546 and a grant from Agilent Technologies.
email: Nischal.Piratla@colostate.edu, Abhijit.Bare@colostate.edu, Anura.Jayasumana@colostate.edu, Rick_Whitner@agilent.com.

Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN'03)
0742-1303/03 $ 17.00 © 2003 IEEE

Let L be the packet size and. C_i be the capacity of the i^{th} hop. D_i is the dispersion as seen at the end of hop i. Consider the cases $C_2 > C_1$ and $C_2 < C_1$ separately where C_1 and. C_2 are capacities of hops 1 and 2 respectively. All the possible dispersions are illustrated in figure 1. It can be shown that when the packet moves to the n^{th} hop, i.e., the last node, we have

$$D_n = d_n + \max (L/C_n, \ldots, L/C_2, L/C_1) - K_n.$$

The computation of K_n increases in complexity as as n increases, but it is always less than d_n [7]. The term d_n is the difference of network delays of packets, i.e. the jitter 'J' between these packets corresponding to the CT and queuing delays. We propose estimating the jitter component J using a set of packets that are spaced so that the bottleneck capacity does not affect the gap between them. The above argument can be extended for a packet train instead of a packet pair with N packets with measured dispersion $D_{n(N)}$ given by $B = (N-1) L/ (D_{n(N)} - J)$. As the cross traffic increases and N increases, $d_n >> K_n$.

3. Captool

Captool sends a train of UDP packets from measurement source to measurement destination. It contains a preamble, a pre-train and an actual-train. Preamble, 20% of the length of the complete train, is used for any initial packet transfer issues. Pre-train and actual-train share the remaining 80% of the packets equally. In pre-train, the packets are sent sufficiently apart that the previous packet does not push the next packet backwards due to its own dispersion in a bottleneck link. Therefore, the jitter between these packets is purely due to CT and queuing delays. Since this pre-train is sent immediately before the actual packet train that is used for dispersion measurement, the average jitter for this pre-train and actual-train are assumed to be approximately constant.

The test suite to test Captool and other tools on known bottleneck capacity links covered different known bottleneck link capacities, hops between the source and the destination, different kinds of CT (UDP/TCP) and varying proportions of CT in the link. The results are shown in tables 1,2 and 3. It can be seen that Captool performs better even with lower capacities and higher CT.

.

4. Conclusions and future work

Captool is a new capacity measurement tool that uses a jitter compensation technique. In experiments conducted in networks with known bottleneck capacities between 256Kbps to 100Mbps, Captool is observed to perform better than other comparable capacity measurement tools. Within Captool itself, inaccuracies can be introduced due to application level implementation issues such as time stamping and delays due to context switching. Kernel level implementations to address some of these issues are being pursued.

Table 1. Type of CT Vs. Estimated Capacity for 10Mbps bottleneck capacity

Tool/CT	100% TCP	100% UDP	85%TCP +15%UDP
Captool	9.86Mbps	9.86Mbps	9.86Mbps
Pathrate	9.75Mbps	9.75Mbps	9.75Mbps

Table 2. 10Mbps bottleneck link with only 3 hops

Additional CT (Mbps)	Tool	Execution Time (sec)	Reading (Mbps)
0	Captool	31	9.65
	Pathrate	16	9.65
	SProbe	0.06	4.3
2	Captool	32	9.61
	Pathrate	16	9.55
	SProbe	0.06	4
4	Captool	32	9.75
	Pathrate	16	9.55
	SProbe	0.07	4.5

Table 3. Cable modem uplink as the bottleneck capacity link

Additional CT (Mbps)	Tool	Execution Time (sec)	Reading (kbps)
100kbps	Captool	242	257.76
	Pathrate	> 1500	850
	SProbe	0.2	1900
150kbps	Captool	330	264
	Pathrate	> 1800	1.15
	SProbe	0.15	1140

5. References

[1] V. Jacobson, "Congestion Avoidance and Control," in *Proceedings ACMSIGCOMM*, Sept. 1988, pp. 314–329.

[2] C. Dovrolis, P. Ramanathan and D. Moore. "What do packet dispersion techniques measure?" Proc. INFOCOM 2001, pp. 905-914.

[3] K. Lai and M. Baker, "Nettimer: A Tool for Measuring Bottleneck Link Bandwidth", *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, March 2001.

[4] S. Saroiu, P. K. Gummadi, S. D. Gribble, "SProbe" , 2002, http://sprobe.cs.washington.edu/.

[5] R. L. Carter and M. E. Crovella, "Measuring Bottleneck Link Speed in Packet-Switched Networks," *Performance Evaluation*, vol. 27-28, 1996. pp. 297–318.

[6] V. Paxson, *Measurements and Analysis of End-to-End Internet Dynamics*, Ph.D. thesis, University of California, Berkeley, Apr. 1997.

[7] N. Piratla, Ph.D. Thesis, in progress.