Estimating of Link Structure and Link Bandwidth.

Khunkitti Akharin*, and Limpattanasiri Wisit**
*Faculty of Information Technology, King Mongkut's Institute of Technology
(Tel : +66-2-737-2551; E-mail: akharin@it.kmitl.ac.th)
**Faculty of Information Technology, King Mongkut's Institute of Technology
(Tel : +66-2-737-2551; E-mail: wisitlim@hotmail.com)

Abstract: Over the last decade the research of end-to-end behavior on computer network has grown by orders but it has few researching in hop-by-hop behavior. We think if we know hop-by-hop behavior it can make better understanding in network behavior. This paper represent ICMP time stamp request and time stamp reply as tool of network study for learning in hop-by-hop behavior to estimate link bandwidth and link structure. We describe our idea, experiment tools, experiment environment, result and analysis, and our discussion in our observative.

Keywords: Link Bandwidth, Network Estimation, Network Behavior, and Multiple-Link.

1. INTRODUCTION

Over the last couple of years we have been experimenting with design and implementation of flow control techniques at the transport layer [1, 3]. Throughout this work, like most researchers in the field, we assumed that the primary cause for losses is congestion in the network, primary cause for variability in round-trip time is cross traffic, and that the servers are work conserving [7, 9]. Our experience with Internet [2,11] indicates that several of these assumptions may not hold.

Many studies of the network behavior have appeared in the literature in resent years. Heimlich [6] has studied traffic on NSFNET, and has reported that the traffic can be characterized using a *packet train model*. Caceres *et al* [5] have done trace analysis of TCP/IP wide-area internetwork traffic. They have characterized conversations by number of bytes, duration, number of packets, etc. Paxson [10] has reported aggregate characteristics of TCP conversations for a month, and studied distributions of amount of data transferred, network bandwidth used, conversation lifetimes etc. Cabrera *et al* [4] have studied performance achieved by user processes in Ethernet based environments. Mills [8] has studied roundtrip transit delays over Internet as a function of packet length, and the effectiveness of TCP retransmission-timeout algorithm. Zhang *et al* [12] have used simulation to study network dynamics. D. Sanghi *et al* [13] describe a simple experiment designed to capture end-to-end behavior of the Internet. Bolot [14] analyze the end-to-end packet delay and loss behavior in the Internet with Sanghi's Tool [13]. He can detect some network problem in his experimental. Leland *et al* [15] underlying mathematical and statistical properties of self-similarity and their relationship with actual network behavior. Kevin Lain *et al* [16] describe a deterministic model of packet delay to measuring link bandwidth.

In order to better understand the characteristics of network paths we conducted a series of estimates link structure and link bandwidth. In this report, we present the results of some of these estimations. The results presented in this report agree with Self-Similar of Ethernet traffic and we founded link bandwidth of multiple-link is serve with one physical link (port) bandwidth.

The rest of the paper is organized as follows. In Section 2, we describe our idea and method of estimating. In Section 3, we describe experimental setup and series of experiments. In Section 4, we outline result of experimental and analysis. In Section 5, we talking about observation and discussion in result and analyze. Section 6 concludes the paper.

2. METHOD OF ESTIMATING

We use specific series of controlled packets to estimating. First we should describe our notations.

LB : Link Bandwidth.
T_{Ri} : Receive time of packet *i.*
T_D : Delay time.
$T_{D(A,B)}$: Delay time between node A and node B.

Looking to Figure1 (A) is idea of packet series to estimate link bandwidth. We send 2 packets of smallest size too close together to target link. We assume there are cross network together and link bandwidth can estimate by equation $LB = Packet\ Size\ /\ minimum\ T_D$ with $T_D = T_{R2} - T_{R1}$. If the TestPC dose not on target link, can estimate link bandwidth by $LB = Packet\ Size\ /(minimum\ T_{D(A,N+1)} - minimum\ T_{D(A,N)})$. With this idea we do not have problem about timing between nodes, so do not need time synchronization between nodes

Looking to Figure1 (B) is idea of packet series to detect link structure is it multiple-link or not. We send 1 packet of biggest size and 2 packets of smallest size too close together to target link. We assume there are cross network together and link structure can estimate by receive time. If receive time of small packet size is less than big packet size ($T_{R2} < T_{R1}$) that shown 2 packets are transmit via difference port (server), we can assume target link is multiple-link. Else target link is normal link.

Looking to Figure1 (C) is idea of packet series to estimate number of port in multiple-link. After detected target link was multiple-link we will do this step. We send series of packet same idea of estimating link structure. We will increase packet of biggest size one by one until receive time of small packet is more than big packet. We can estimate number of port in multiple-link with number of big packet. We just make sure every packet in series was sending in effective time. The effective time can calculate with equation *Effective Time = Biggest Packet Size / Link Bandwidth.*

This section we show our ides and method of estimating link bandwidth, link structure, and number of port in multiple-link.

Figure1: Series of estimating packet.

3. EXPERIMENTAL SETUP

This section we describe our study tools, hardware, and software, experimental environment configuration, and series of experiments.

3.1 Experiment tools

Our tool to study link behavior is ICMP time stamp request and ICMP time stamp reply [17] with out requiring any modification to the kernel implementation of network node because of they are following standard of Internet protocol suite [18 – 20]. ICMP time stamp request and reply packet format is shown in Figure 2. The Originate Timestamp is the time the sender last touched the message before sending it, the Receive Timestamp is the time the echoer first touched it on receipt, and the Transmit Timestamp is the time the echoer last touched the message on sending it. The data received (a timestamp) in the message is returned in the reply together with an additional timestamp. The timestamp is 32 bits of milliseconds since midnight UT.

Experiment hardware, we use Dell® OPTIPLEX GX270 series to be TestPC. TestPC specification is Intel® Pentium4 2.8 GHz with Hyper Threading central processing, 1024 MB of RAM, 80 GB of harddisk, and Intel® PRO/1000MT 10/100/1000 Mbps full duplex on board NIC; TestPC operate with *Microsoft Windows® XP Service Pack 2*. We use 2 of Cisco® Catalyst 3550 Multilayer Switch every port are full duplex operate with Cisco® IOS release 12.2(25)SEB. It can make layer 3 multiple-ling with Link Aggregation Control Protocol [22] maximum 8 ports of same bandwidth (1 – 100000 kilobits). Their name is "Router1" and "Router2".

Experiment software, we use software *hping* version 2 [21] to generate prove packet; software hping can control packet interval-time, packet type, and packet size. The hping need Windows Packet Capture library and we use *WinPCap* version3.0 [25]. We use software *CommView* version 5.0 [24] to capture traffic in network; it can detect time of packet behavior in microsecond (10^{-6} second) scale.

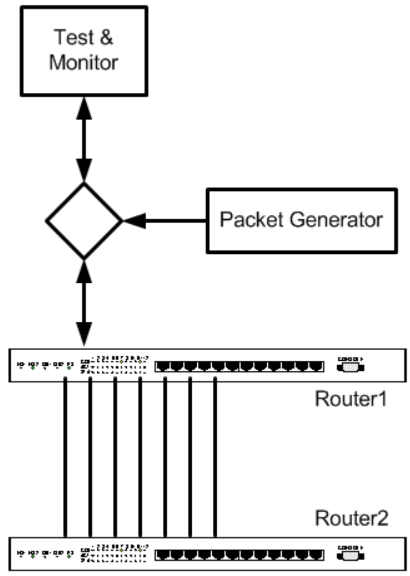

Figure3: Experimental environment.

3.2 Experiment environment

Experiment environment is shown in Figure3. Both of network nodes, we configure every port in multiple-link at interface FastEthernet0/13 - FastEthernet0/20 to be running with 10 kilobits bandwidth. Set bandwidth between Router1 and TestPC via interface FastEthernet0/23 with 100,000 kilobits

On TestPC, we run test mechanism for each experiment and monitor result. After we make sure in our idea, we will add Packet Generator module to simulate network traffic and measurement performance of our mechanism and algorithm in any case of traffic load. All experiments are repeating at least 100 times and use average value to present in this paper.

3.3 Experiment#1 Estimating Link Bandwidth

Objective is to estimate link bandwidth and learning in link bandwidth behavior of multiple-link. We use ICMP timestamp request with 60 bytes packet size send set of packets 100 times and find minimum delay. First we find minimum delay between TestPC and Router1. Second we plug UTP cable only 1 port in multiple-link and estimate link bandwidth with algorithm in Section 2. We increate port in multiple-link via plug UTP cable and estimate link bandwidth each number of ports in multiple-link.

00–07	08–15	16–31
Type	Code	ICMP header checksum
Identifier		Sequence number
Originate timestamp		
Receive timestamp		
Transmit timestamp		

Figure 2: ICMP type 13 and 14 packet format.

3.4 Experiment#2 Detecting Multiple-Link

Objective is to detect multiple-link. We use the minimum size of Ethernet prove packet is 60 bytes and the maximum size of Ethernet prove packet is 1,514 bytes. First packet is 1,514 bytes size; second packet and third packet are 60 bytes size. Every prove packet must have send in *1,514 bytes / Link Bandwidth*. We experiment both of normal link and multiple-link.

In link bandwidth size 100000 kilobits need processing time to transmit 1 packet size 1514 byte at lease

$$\frac{1514 \times 8}{100000 \times 1024} = 0.00011828125 \text{ second.}$$

In link bandwidth size 100000 kilobits need processing time to transmit 1 packet size 60 byte at lease

$$\frac{60 \times 8}{100000 \times 1024} = 0.0000046875 \text{ second.}$$

In link bandwidth size 10 kilobits need processing time to transmit 1 packet size 1514 byte at lease

$$\frac{1514 \times 8}{10 \times 1024} = 1.1828125 \text{ second.}$$

In link bandwidth size 10 kilobits need processing time to transmit 1 packet size 60 byte at lease

$$\frac{60 \times 8}{10 \times 1024} = 0.046875 \text{ second.}$$

To detect link structure we send 1 packet size 1514 bytes and 2 packets size 60 byte from TestPC to Router2 to detect link structure between Router1 and Router2. Their will already at Router1 in 0.000118228125 + (0.0000046875 * 2) = 127.603125 microsecond and increase time length by multiple with 2 is 255.20625 microsecond. This calculate time is less than 1 packet size 1514 bytes transmit in link bandwidth size 10 kilobits (1.1828125 second)

3.5 Experiment#3 Estimating Number of Ports in Multiple- Link

Objective is to estimate number of active ports in multiple-link. We use the minimum size of Ethernet prove packet is 60 bytes and the maximum size of Ethernet prove packet is 1,514 bytes. We start with one big packet and fallow with 2 small packet. We increase number of big packet with 1 until receive time of small packet is more than last big packet. Every prove packet must have send in *1514 bytes / Link Bandwidth*.

By our experimental environment, N is maximum number of packet size 1514 to estimate number of port is not more than 5002 packet, can calculate by equation

$$N < \frac{\frac{1.1828125}{2} - (2 \times 0.0000046875)}{0.000118228125}$$

But with out hardware can make maximum 8 ports for multiple-link.

3.6 Experiment#4 Network Load Factor Effect to Result.

Objective is to measure mechanism performance when have load in network. We use packet generator to generate packet size 1514 bytes with difference sending rate into network and use our algorithm to estimate link bandwidth and link structure, measuring result and comparing.

4. RESULTS AND ANALYSIS

This section we shown result of our experimental. After we do experiment we have found timestamp value from ICMP mechanism is not good enough to study Ethernet behavior because of Ethernet process in microsecond time scale but ICMP timestamp value is millisecond time scale. We use time value that measure with software *CommView*, we process in same idea and mechanism by use back turn receive time.

In Figure 4, shown link bandwidth estimation value between Router1 and Router2 that hardware specification is 1 kilobits of bandwidth. We experiment 1 – 8 ports of multiple-link and its result are same. Then we push traffic in to network to make network load from 1% to 50% network load and experimenting. The X-axis is percentage of network load. The Y-axis is bandwidth in unit of kilobits. The estimation value is 8 kilobits.

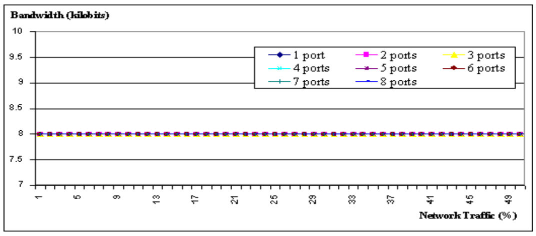

Figure 4. Link bandwidth estimation result

In Figure 5, shown result of link structure detecting of normal link between TestPC and Router1. In 100 times of experimental, we can detect link structure with 100 percent of correct answer. Result values "1" is multiple-link and value "0" is normal link.

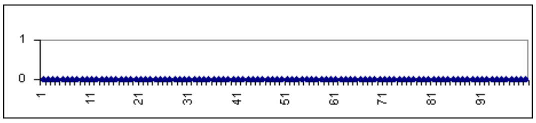

Figure 5. Link structure detection result of normal link

In Figure 6, shown result of link structure detecting of multiple-link between Router1 and Router2. In 100 times of experimental, we can detect link structure with 100 percent of correct answer. Result values "1" is multiple-link and value "0" is normal link.

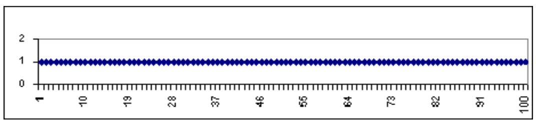

Figure 6. Link structure detection result of multiple-link

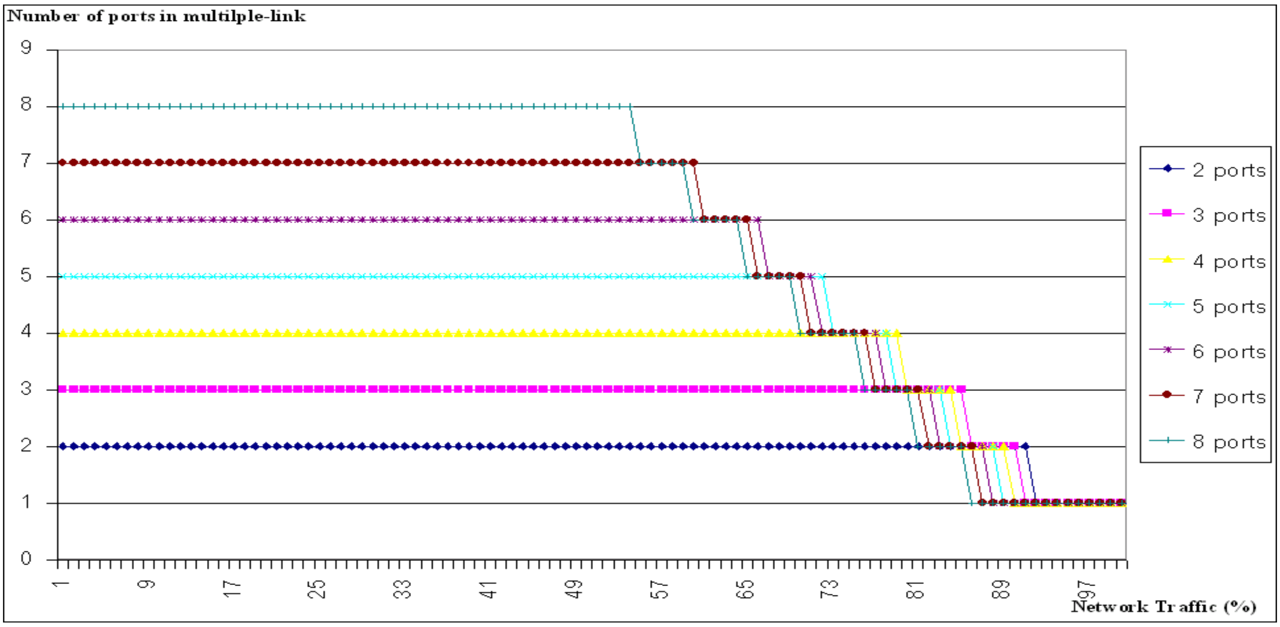

Figure 7. Estimating number of port in multiple-link

In Figure 7, shown result of number of ports in multiple-link estimation. The X-axis is number of ports estimation result. The Y-axis is network load generate by packet generator. We show the average estimating result from 100 times experimental for each number of ports in multiple-link and each network load.

5. OBSERVATIONS AND DISCUSSION

In this section, we present some preliminary observations based on the results of our experiments. Cleary, these are initial observations and the discussion reflects our current thinking.

5.1 Observation

1. Traffic's behavior of Ethernet network is Self-Similar.
2. Minimum of sending gap is function of packet size and link speed.
3. Timing and time synchronization between network node is a important problem for studying of network behavior.
4. ICMP timestamp values in millisecond time scale are out of date because of present network link, network node and network edge are process in microsecond time scale.
5. Estimation of link bandwidth is not fit to real physical link property that means more than transmission delay in measurements delay.

5.2 Discussion

1. The computer that use to running experimental process is should be very high performance.
2. Prove packet generator need ability to control sending interval time in microsecond time scale.
3. Error of estimating link bandwidth is effect from error of delay measurement.

6. CONCLUSION

This paper present a network studying in case of multiple-link by represented idea and design of algorithm to estimating link bandwidth, detecting multiple-link, and estimating number of ports in multiple-link. Our experimental and result was process and monitoring on a setting up network environment with only 3 network nodes are TestPC, Routerr1, and Router2. but we was found some thing for our work.

Our future work is to apply and modified our algorithm to be done for real network.

ACKNOWLEDGMENTS

Special thanks to Dr. S. Chantaboon and Dr. C. Nopporn guide me to partitioning experiments and how to expect goal. Dr. P. Chotipate from network research laboratory provide Cisco Catalyst 3550 Multilayer Switch. P. Nattachote, my friends provided interesting points in experiments. Information Technology Faculty and IT staffs provide IT facilities, and the anonymous referees provide valuable feedback.

REFERENCES

[1] Ashok K. Agrawala and Dheerah Sanghi. "Design and evaluation of an adaptive flow control scheme" In *Proceedings of IEEE INFOCOM '92,* Florence, Italy, May 1992.

[2] Ashok K. Agrawala and Dheerah Sanghi. "Network dynamics: an experimental study of the Internet" In *Proceedings of Globecom '92,* Orlando, FL, December 1992.

[3] J-C. Bolot and A.U. Shanker. "Analysis of a fluid approximation to flow control dynamics" In *Proceedings of IEEE INFOCOM '92,* Florence, Italy, May 1992.

[4] Luis-Felipe Cabrera, Edward Hunter, Michale K. Karels, and David A. Mosher. "User-process connumication performance in networks of computers" *IEEE Transactions on Software Engineering,* 14:38-53, January 1988.

[5] R. Caceres, P.B. Danzing, S. Jamin, and D.J. Mitzel. "Characteristics of wide-area TCP/IP conversations" In *Proceedings of ACM SOGCPMM '91*, pages 101-112, Zurich, Switzerland, September 1991.

[6] Steve A. Heimlich. "Traffic characterization of the NSFNET nation backbone" In *Proceedings of USENIX Winter Conference '90,* Washington DC, January 1990.

[7] A. Mankin. "Random drop congestion control" In *Proceddings of ACM SIGCOMM '90,* pages 1-7, Philadelphia, Pennsylvania, September 1990.

[8] D. L. Mills. "Internet delay experiments" Request for Comment: RFC-889, Network Information Center, SRI International, January 1984.

[9] J. Nagle. "Congestion control in IP/TCP internetworks" Request for Comment: RFC-896, Network Information Center, SRI International, January 1984.

[10] Vern Paxson, "Measurements and models of wide area TCP conversations" Technical Report LBL-30840, Computer Systems Engineering Department, Lawrence Berkeley Laboratory, University of California, Berkeley, CA, May 1991.

[11] Dheeraj Sanghi, M. C. V. Subramaniam, A. U. Shanker, Ó. Gudmundsson, and P. Jalote. "A TCP instrumentation and its use in evaluating roundtrip-time estimators" *J.of Internetworking: Research & experience,* 1920:77-99, December 1990.

[12] Lixia Zhang and David D. Clark. "Oscillating behavior of network traffic: A case study simulation" *J. of Internetworkign: Research & Experimence,* 1(2):101-112, December 1990.

[13] D. Sanghi, A. K. Agrawala, Ó. Gudmundsson, and B.N. Jain. "Experimental Assessment of End-to-end Behavior on Internet" In *Proceedings. IEEE Infocom '93,* San Fransisco, CA, pp.867-874, March 1993.

[14] Jean-Chrysostome Bolot. "End-to-End Packet Dealy and Loss Behavior in the Internet" In *Proceedings ACM SIGCOMM '93,* Ithaca, N.Y., USA, September 1993.

[15] Will E. Leland, Walter Willinger, Murad S. Taqqu and Daniel V. Wilson. "On the Self-Similar Nature of Ethernet Traffic (Extended Version)", *IEEE/ACM Transactions on Networking,* vol. 2, no. 1, pp. 1-15, Feb 1994.

[16] Kevin Lai and Mary Baker, "Measuring Link Bandwidths Using a Deterministic Model of Packet Delay",

[17] RFC792: INTERNET CONTROL MESSAGE PROTOCOL, http://www.faqs.org/rfcs/rfc792.html

[18] RFC791: INTERNET PROTOCOL, http://www.faqs.org/rfcs/rfc791.html

[19] RFC 1122: Requirements for Internet Hosts – Communication Layer, http://www.faqs.org/rfcs/rfc1122.html

[20] RFC 1812: Requirements for IP Version 4 Routers, http://www.faqs.org/rfcs/rfc1812.html

[21] Software *hping version2,* http://www.wiki.hping.org

[22] Link Aggregate Standard, IEEE 802.3ad, , http://www.ieee802.org/3/ad/index.html

[23] Catalyst 3550 Multilayer Switch Software Configuration Guide. Cisco Systems, Inc. USA., 2005 http://www.cisco.com

[24] Software *Commview version 5.0*, http://www.tamos.com/download/main/

[25] Software library *WinPCap version 3.0*, http://winpcap.polito.it/