Inline Network Measurement: TCP with an Built-in Measurement Technique

Cao Le Thanh Man, Go Hasegawa and Masayuki Murata
Graduate School of Information Science and Technology, Osaka University
1-3, Yamadagaoka, Suita, Osaka 560-0871, Japan
Phone: +81-6-6850-6863, Fax: +81-6-6850-6868
E-mail: mlt-cao, hasegawa, murata@ist.osaka-u.ac.jp

Foreword

We introduce a novel mechanism for actively measuring available bandwidth along a network path. The new measurement mechanism is built in the TCP sender; it exploits data packets transmitted in a TCP connection (inline measurement), instead of adding probe traffic to the network. The TCP sender with the proposed measurement mechanism adjusts the transmission intervals of data packets, then estimates available bandwidth of the network path between sender and receiver utilizing the arrival intervals of ACK packets. Simulations show that the new measurement mechanism does not degrade TCP data transmission performance and yields acceptable measurement results in intervals.

Extended Abstract

Information concerning bandwidth availability in a network path plays an important role in adaptive control of the network. Network transport protocols can use such information to optimize link utilization or improve transmission performance. In particular, for optimal route selection in service overlay networks, fast and accurate information on available bandwidth is needed. Available bandwidth information is also used in network topology design and is a key factor in network troubleshooting.

Available bandwidth can be measured at routers within a network. This approach may require a considerable change to network hardware and is suitable for network administrators only. Some *passive* measurement tools can collect traffic information at some end hosts for performance measurements, but this approach requires a relatively long time for data collection and bandwidth estimation. Exchanging probe traffic between two end hosts to find the available bandwidth along a path (an *active* measurement) seems the more realistic approach and has attracted much recent research.

Sending extra traffic into the network is the common weakness in all active available bandwidth measurement tools. For example, Pathload [1] generated between 2.5 to 10 MB of probe traffic per measurement. For routing in overlay networks, or adaptive control in transmission protocols, these measurements may be repeated continuously and simultaneously from numerous end hosts. In such cases, the probes will create a large amount of traffic that may damage other data transmission in the network as well as degrade the measurement itself.

We propose an active measurement method that does not add probe traffic to the network, with the idea of "plugging" the new measurement mechanism into an active TCP connection (*inline measurement*). That is, data packets and ACK packets of an TCP connection are utilized for the measurement. This method has the advantage of requiring no extra traffic to be sent to the network.

The idea of inline measurement has previously appeared in traditional TCP. To some extent, traditional TCP can be considered a tool for measuring available bandwidth because of its ability to adjust the congestion window size to achieve a transmission rate appropriate to the available bandwidth. However, the estimation is insufficient and inaccurate because it is a measure of *used* bandwidth, not *available* bandwidth. TCP Westwood [2] apply a passive method in which the sender checks ACK arrival intervals to infer available bandwidth. Because the method observes only ACK arrival intervals, changes in available bandwidth cannot be detected quickly.

In this study, we introduce ImTCP (Inline measurement TCP), a Reno-based TCP that deploy active method for inline measurement. The ImTCP sender does not only observe ACK packet arrival intervals, but also actively adjusts the transmission interval of data packets. When the sender sends data packets, it also adjusts the packet transmission intervals, just as active measurement tools do with probe packets. When the corresponding ACK packets return, they are considered to be the echoed packets of probe traffic. The sender then utilizes the arrival interval of these packets to calculate the available bandwidth. The sender thus collects more information for a measurement and improved accuracy can be expected.

The ImTCP sender performs periodic measurements at short intervals, on the order of several RTTs. During every measurement, the sender searches for the available bandwidth only within a given *search range*. The search range is a range of bandwidth that is expected to include the current available bandwidth and is calculated from the previous measurement results. By introducing the search range, we can avoid sending probe packets at an extremely high rate. We can also keep the number of probe packets for the measurement quite small. The search range is divided into multiple sub-ranges of indentical width of bandwidth. For each of the sub-range of the bandwidth, the sender transmits simultaneously a group of packets (packet stream), of which the transmission rate vary to cover the sub-range. The sender then check to see if an increasing trend exists in the transmission delay of each stream when the echoed packets arrives at the sender host. Using the characteristic that the increasing trend of the transmission delay in a stream indicates that the transmission rate of the stream is larger than the current available bandwidth of the network path [1], the sender infers where the available bandwidth is in the search range. The algorithm is described in detail in [3].

We insert a measurement program into the sender program of TCP Reno to create ImTCP sender. The measurement program is located at the bottom of the TCP layer. When a new data packet is generated at the TCP layer and is ready to be transmitted, the packet is stored in an intermediate FIFO buffer. The measurement program decides the time at which to send the packets in the buffer. On the other hand, when an ACK packet arrives at the sender host, the measurement program records its arrival time and passes the packet to the TCP layer for TCP protocol processing. When ImTCP performs a measurement, the program waits until the number of packets in the ImTCP is sufficient to form a packet stream, and then sends the stream at the transmission rate determined by the measurement algorithm. The program sends all streams required for a measurement and then calculates the measurement result from the time intervals of the corresponding ACK packets. While waiting for the ACK packets, the program passes all data packets immediately to the IP layer. When the window size is sufficiently large, the program creates all streams required for a measurement for each RTT. When the current window size is smaller than the number of packets required for a packet stream, the program stores no data packets, which means that ImTCP does not perform measurement. The measurement program does not require any special changes in the TCP receiver program, except that a probe packet must be sent back for each received packet. Therefore, delayed ACKs must be disabled at the TCP receiver, otherwise ImTCP will not perform measurement properly.

Our simulation in ns-2 uses the network model described in Figure 1. The ImTCP sender sends data to the receiver and, at the same time, measures the available bandwidth of the share link. The background traffic is made up of UDP packet flows. During the simulation, the background traffic is changed so that the available bandwidth is changed, as showed by "A-bw" line in Figure 2. The line "ImTCP" in Figure 2 shows the average

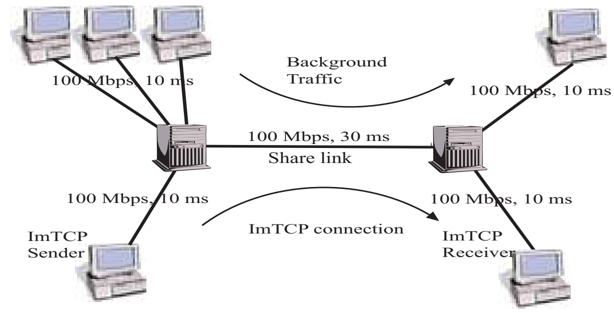

Figure 1: Network model

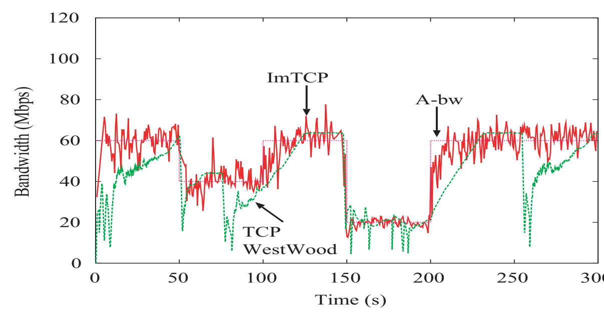

Figure 2: Average measurement results

measurement results for every 0.5 sec. For comparison, we also show the measurement results of TCP Westwood (using the latest available version [2]) under the same network conditions. Figure 2 indicates that when the available bandwidth increases suddenly, TCP Westwood yields results that are lower than the true value because the data transmission rate cannot adjust as rapidly and needs time to ramp up because of the self-clocking behavior of TCP. In contrast, the measurement results of ImTCP can reflect well the changes in the available bandwidth.

References

[1] M. Jain and C. Dovrolis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput," in *Proceedings of ACM SIGCOMM 2002*, Aug. 2002.

[2] M.Gerla, B.Ng, M.Sanadidi, M.Valla, R.Wang, "TCP Westwood with adaptive bandwidth estimation to improve efficiency/friendliness tradeoffs," *To appear in Computer Communication Journal*.

[3] Cao Man, Go Hasegawa and Masayuki Murata, "A new available bandwidth measurement technique for service overlay networks," in *Proceeding of 6th IFIP/IEEE International Conference on Management of Multimedia Networks and Services Conference, MMNS2003*, pp. 436–448, Sept. 2003.