An Efficient and Loss Tolerant Method for Measuring Available Bandwidth

Victor Fajardo, Yuu-Heng Cheng
Telcordia Technologies, NJ, USA
fajardov@telcordia.com
yhcheng@research.telcordia.com

Gautam Parmar
Toshiba America Research, Inc., CA, USA
parm_gautam@yahoo.com

Yoshihiro Ohba
Toshiba R&D Center, Kawasaki, Japan
yoshihiro.ohba@toshiba.co.jp

Abstract—Available bandwidth in a network path is an important performance criterion for evaluating the network performance. In this paper, we propose a new algorithm called Single Burst (SB) for measuring the available bandwidth between end hosts. Additionally, we provide mathematical analysis to derive approximate upper bounds on amount of exchanged data and measurement latency for SB. Finally, we present a comparison of measurement latency, accuracy, usability, and network overhead among an implementation of SB and two existing available bandwidth measurement tools, i.e., Iperf and pathChirp. As a result, we conclude that SB measures the available bandwidth even under the condition of packet losses, with default settings, with faster response time, and without generating excessive test traffic.

Index Terms—network capacity, measurement, network monitoring, performance.

I. INTRODUCTION

As more link layer technologies are available lately, a demand for user devices equipped with multiple network interfaces supporting different link layer technologies has been growing. It is crucial for such multi-interface devices to be able to determine which network (or networks) to use based on user requirements and network performance without user intervention. In this paper, we focus on a method to support such decisions by obtaining the unused capacity or available bandwidth of a network. This method should promptly and accurately measure the available bandwidth for different network types in order to reduce the network selection latency and may hence improve the overall user experience. Moreover, the method should not inject excessive traffic into the network during the measurement. There are several existing methods to measure the available bandwidth such as Iperf [1], pathload [2], TOPP [3], pathChirp [4], etc. Existing tools, however, have at least one of the following drawbacks.

First, response time for the measurement, i.e. measurement latency, is too large. It is known that available bandwidth measurement methods that use TCP show a high measurement latency (e.g., more than 10 seconds) due to the “slowstart” flow control behavior of TCP [5]. Though Pathload and pathChirp uses UDP, they require a number of test iterations to adjust the inter-transmission intervals of test packets, which results in higher measurement latency while they require fewer amount of data exchanged.

Second, existing UDP-based available bandwidth measurement methods, such as pathChirp and iPerf in UDP mode, may reduce accuracy in an environment where test packets can get lost due to, e.g., lower SNR (Signal to Noise Ratio) or interference of a wireless link and network congestion. Therefore they may not be suitable for all network types. TCP-based methods are robust against packet loss but they have the measurement latency issue as described earlier.

Finally, too many tunable parameters decrease the usability of the measurement method. Some methods require the user to provide parameters under different network conditions to be able to achieve a certain level of measurement accuracy. For example, pathChirp requires a lower bound and an upper bound for the measured available bandwidth as well as the packet size as input parameters. The use of the pre-configured default values in Iperf can result in either reduced measurement accuracy, or increased measurement latency or excessive amount of exchanged data.

To overcome these drawbacks, we propose a new UDP-based available bandwidth method called Single Burst (SB). SB is a variant of the packet train probing scheme [6] and designed to be robust against the condition of high packet losses, and has short measurement latency without requiring input parameters from users. Additionally, SB does not require excessive amount of test data.

We use the following terms in the rest of the paper. A source node is the active node for measuring the available bandwidth. A target node is the passive node for measuring the available bandwidth. A forward path is a path from the source node to the target node. A backward path is a path from the target node to the source node. A roundtrip path is the concatenation of the forward path and the backward path where the concatenated path is originated and terminated at the source node. The roundtrip available bandwidth is useful for interactive real-time applications such as online games. Directional available bandwidth is the available bandwidth of either a forward path or a backward path. MTU refers to Maximum Transmission Unit. RTT refers to Round-Trip Time.

The remainder of the paper is organized as follows. Section II describes the SB algorithm, the state machine used for packet cluster formation, and the SB protocol between a source and a target node. Section III provides mathematical analysis of the algorithm. Section IV shows performance comparison among Iperf, pathChirp and SB. Section V concludes the paper.
II. SB ALGORITHM

A. Algorithm Details

SB measures directional and roundtrip available bandwidth. In SB, one node transmits one or more test packets of size \( L \) (octets) including IP header to the other node at rate \( R_t \) (b/s) and then waits for test receipt packets returned from the other node. A pair of test and test receipt packet is matched by the sequence number carried in each packet.

Figure 1 depicts the general message exchange between the test packet sender and receiver. The sender transmits at least \( N_{\text{min}} \) octets of test packets and stops transmission if a test receipt packet has been received after transmitting \( N_{\text{min}} \) octets (at time of S1) or when transmitted \( N_{\text{max}} \) octets (at time of S2). That is, the amount of test packets to transmit is dynamically determined based on the RTT and the transmission rate \( R_t \). If \( R_t \) is not specified, the test packets are transmitted at the highest rate afforded by the sender. In the case of backward path available bandwidth test, \( R_t \) may be specified by the source node.

Figure 1: Basic Sequence of Single Burst Measurement

The sender and receiver of test packets as well as the entity that computes the available bandwidth depends on the type of test as shown in Table 1. In the case of directional available bandwidth test, i.e. forward path test or backward path test, test receipt packets do not contain a payload in order to load only the path on which test packets are traversing. In the case of roundtrip path available bandwidth test, test receipt packets contain the same amount of payload as the test packets to load both forward and backward paths.

Table 1: Roles of Source and Target Nodes

<table>
<thead>
<tr>
<th>Test Type</th>
<th>Test packets sender</th>
<th>Computing node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward path</td>
<td>Source node</td>
<td>Target node</td>
</tr>
<tr>
<td>Backward path</td>
<td>Target node</td>
<td>Source node</td>
</tr>
<tr>
<td>Roundtrip path</td>
<td>Source node</td>
<td>Source node</td>
</tr>
</tbody>
</table>

In order to maintain accuracy of measurement under the condition of packet losses, we define a packet cluster as a set of at least two delivered test packets whose sequence numbers are continuous. Packet clusters are arranged in ascending order of smallest sequence number in each cluster. For example, let \( p_i \) denote the test packet whose sequence number is \( i \); let \( c_k \) denote \( k \)-th packet cluster. Assuming test packets were observed to be delivered in the following order \( p_1, p_2, p_3, p_5, p_6, p_{10}, p_{11} \), three clusters are formed as \( c_1 = \{ p_1, p_3, p_2 \} \), \( c_2 = \{ p_5, p_6 \} \), and \( c_3 = \{ p_{10}, p_{11} \} \). Packet \( p_{10} \) does not form a packet cluster. More detailed explanation on packet clusters is given in Section B.

The available bandwidth \( B = \sum_{k=1}^{C} 8(m_k - 1)L \sum_{k=1}^{C} d_k \) (b/s) where \( m_k \) denotes the number of test packets in \( k \)-th packet cluster (\( m_k \geq 2 \)), \( d_k \) denotes the inter-delivery time in seconds between the first and last test packets in \( k \)-th packet cluster, and \( C \) denotes the total number of packet clusters. For roundtrip available bandwidth tests, a test packet is considered as delivered when the corresponding test receipt packet is received by the source node. If path MTU is known, \( L \) is the path MTU. Otherwise \( L \) is set to the minimum MTU, i.e., 576 octets for IPv4 and 1280 octets for IPv6.

In order to minimize the impact of test receipt packets on the measured available bandwidth for directional tests \([7]\), the receiver of test packets may selectively transmit test receipt packets. This feature is referred to as Selective ACK (SACK). For example, a test receipt packet may be transmitted for every 10 test packets received instead of for each test packet.

B. Packet Cluster State Machine

We describe an algorithm to form a list of packet clusters from an array of test packet delivery times in this section. The complexity is \( O(N) \) in terms of the number of test packets.

Let \( a[] \) denote an array where \( a[i] \) stores the delivery times of \( i \)-th test packet. A delivery time of zero in the array indicates that the corresponding test packet was not delivered. Let \( last \) denote the sequence number of the last test packet. Let \( \text{add_cluster}(\text{int } \min_a, \text{int } \max_a, \text{int } n) \) be a procedure that creates a new packet cluster for which the minimum delivery time, the maximum delivery time and the number of packets in the cluster is given by \( \min_a \), \( \max_a \) and \( n \), respectively.

The packet cluster detection state machine is invoked at the end of a SB test. It consists of three states Out of Cluster, Eligible Cluster and In Cluster. State Out of Cluster is the initial state and indicates that \( i \)-th test packet was not delivered. State In Cluster indicates that \( i \)-th test packet belongs to the current packet cluster. State Eligible Cluster indicates that \( i \)-th test packet belongs to a new packet cluster if and only if \( i \)-th and \( (i+1) \)-st test packets were delivered and \( (i-1) \)-st test packet was not delivered. The state machine is depicted in Figure 2. Each arrow represents a state transition and a label (“x/y”) associated with each transition indicates a pair of condition x and action y. A state transition that has a null condition indicates that the state transition unconditionally happens. A state transition that has a null action indicates that no action is taken for the state transition.
The SBP sequence for each test type is shown in Figure 4. The source node starts and stops the test for forward path test by sending Data message and end with Finish message. If no SACK is used, the target node will respond with Data-ACK messages for each Data message received. After receiving the Finish message, the target node then sends Report message along with the computed result. For backward path test, the source node sends Start message to trigger the target node to start sending Data message. It sends Data-ACK message for each received Data message and computes the result after receiving the Finish message. In the roundtrip path test, the source node sends Echo message and ends with Finish message. The target node responds with Echo-Reply and Finish messages. SB roundtrip path test does not require any state to be maintained on the target node.

The SBP sequence for each test type is shown in Figure 4. The source node starts and stops the test for forward path test by sending Data message and end with Finish message. If no SACK is used, the target node will respond with Data-ACK messages for each Data message received. After receiving the Finish message, the target node then sends Report message along with the computed result. For backward path test, the source node sends Start message to trigger the target node to start sending Data message. It sends Data-ACK message for each received Data message and computes the result after receiving the Finish message. In the roundtrip path test, the source node sends Echo message and ends with Finish message. The target node responds with Echo-Reply and Finish messages. SB roundtrip path test does not require any state to be maintained on the target node.

The SBP sequence for each test type is shown in Figure 4. The source node starts and stops the test for forward path test by sending Data message and end with Finish message. If no SACK is used, the target node will respond with Data-ACK messages for each Data message received. After receiving the Finish message, the target node then sends Report message along with the computed result. For backward path test, the source node sends Start message to trigger the target node to start sending Data message. It sends Data-ACK message for each received Data message and computes the result after receiving the Finish message. In the roundtrip path test, the source node sends Echo message and ends with Finish message. The target node responds with Echo-Reply and Finish messages. SB roundtrip path test does not require any state to be maintained on the target node.

The SBP sequence for each test type is shown in Figure 4. The source node starts and stops the test for forward path test by sending Data message and end with Finish message. If no SACK is used, the target node will respond with Data-ACK messages for each Data message received. After receiving the Finish message, the target node then sends Report message along with the computed result. For backward path test, the source node sends Start message to trigger the target node to start sending Data message. It sends Data-ACK message for each received Data message and computes the result after receiving the Finish message. In the roundtrip path test, the source node sends Echo message and ends with Finish message. The target node responds with Echo-Reply and Finish messages. SB roundtrip path test does not require any state to be maintained on the target node.

The SBP sequence for each test type is shown in Figure 4. The source node starts and stops the test for forward path test by sending Data message and end with Finish message. If no SACK is used, the target node will respond with Data-ACK messages for each Data message received. After receiving the Finish message, the target node then sends Report message along with the computed result. For backward path test, the source node sends Start message to trigger the target node to start sending Data message. It sends Data-ACK message for each received Data message and computes the result after receiving the Finish message. In the roundtrip path test, the source node sends Echo message and ends with Finish message. The target node responds with Echo-Reply and Finish messages. SB roundtrip path test does not require any state to be maintained on the target node.
[6] assumes FCFS (First-Come First-Served) as oppose to assumption (ii), making the computed bit rate higher than the available bandwidth because the cross traffic output rate during probing is decreased due to interference by the probe traffic in the same buffer, and for this reason the packet train probing scheme is characterized as a tool to measure the path capacity instead of the available bandwidth in [6]. On the contrary, with assumption (ii), the cross traffic is not affected by the probe traffic. This indicates that the probe traffic is transmitted on the tight link at the rate equal to the available bandwidth. Moreover, SB and any variant of the packet train probing scheme are usable for measuring the available bandwidth even with cross traffic. Strict Priority Queuing (SPQ) with the probe traffic having lower priority than the cross traffic is a queuing discipline for which assumption (ii) holds.

Let \( B \) denote the measured available bandwidth (\( B \leq R_t \)). Let \( T_p \) denote the roundtrip propagation delay. Let \( D \) denote the measurement latency in seconds. Let \( N_t, N_f \) and \( N_r \) denote the total number of transmitted octets for test and test receipt packets, the total number of transmitted octets for test packets and the total number of transmitted octets for test receipt packets, respectively (\( N = N_t + N_f \)). Let \( \lceil \cdot \rceil \) denote the ceiling function which returns the integer value that is the smallest integer not less than \( x \), i.e. \( x \leq \lceil x \rceil < x+1 \). The total octets of transmitted test packets, \( N_t \), is given as follows.

Equation 1 \( N_t = \max \left( \min \left( \left[ R_t T_p / 8L \right] L, N_{\text{max}} \right), N_{\text{min}} \right) \)

Since we assume that there is no packet loss (i.e., \( C=1 \)), the following equation holds.

Equation 2 \( B = 8m_1 - 1)L/d_1 \)

Since \( m_1 = N_t / L \).

Equation 3 \( d_1 = 8(N_t - L)/B \)

Table 3 summarizes the mathematical analysis where \( \bar{D} \) and \( \bar{N} \) denote an upper bound of \( D \) and \( N \), respectively, and \( \bar{N}_t = \max \{ \min (R_t T_p / 8+L, N_{\text{max}}), N_{\text{min}} \} \).

See Appendix for detailed analysis. Since \( T_p \) is independent of \( R_t \), the closer \( R_t \) is to \( B \), the less \( \bar{D} \) and \( \bar{N} \) are. If \( N_{\text{min}} < R_t T_p / 8+L < N_{\text{max}} \), the minimum values of \( \bar{D} \) and \( \bar{N} \) are given when \( R_t = B \).

Table 3: Upper Bounds on Measurement Latency and Transmitted Octets

<table>
<thead>
<tr>
<th>Test Type</th>
<th>( \bar{D} )</th>
<th>( \bar{N} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directional</td>
<td>( 2T_p + w + 8(N_t - L) / B )</td>
<td>( \bar{N}_t(1 + 36 / L) )</td>
</tr>
<tr>
<td>Roundtrip</td>
<td>( T_p + w + 8(N_t - L) / B )</td>
<td>( 2N_t )</td>
</tr>
</tbody>
</table>

IV. PERFORMANCE COMPARISON

Comparison of different available bandwidth measurement methods based on analysis, simulations or actual measurements can be found in several papers such as [6], [8], [9], and [2]. We evaluated the performance of SB by comparing with two different types of available bandwidth measurement methods: Iperf (IP, for short; version 2.0.3) and pathChirp (version 2.4.1). We collected the following data from all three methods:

- Measurement latency (ML): the amount of time used to finish the test.
- Available bandwidth (BW): the measured available bandwidth.
- Variance of average bandwidth (VAR): the standard deviation for the average bandwidth of 100 iterations.
- Number of octets generated (NO): the number of octets generated for the test. For SB tests, only test and test receipt packets were counted.
- Packet Loss (PL): the packet loss ratio.

The performance criteria were evaluated for the following types of networks as shown in Figure 5:

- Ethernet. The source node is connected to a switch via a 100 Mb/s, full-duplex Ethernet link.
- WLAN (Wireless LAN). The source node is connected to an IEEE 802.11g access point.
- EVDO (Evolution-Data Optimized). The source node is connected to a commercial cdma2000 EVDO Rev A network. The source node communicates with the target node over the Internet.

Figure 5: Network configuration for different source network types

Only the EVDO network is connected through the Internet. The Ethernet and WLAN are in a controlled environment where the network contains no other traffic but from the two end hosts. One single test consists of 100 samples, and the result for each criterion is computed by averaging the successful tests of these samples. Between each test, there is a 5 seconds pause to make sure that the system does not contain remaining packets from previous test that will affect the current. Iperf is configured to operate in UDP mode with payload size of 1470 octets. The maximum test duration is 1 second, which is the minimum configurable duration value. We change the Iperf “bandwidth” parameter (which identifies the transmission rate) to the link capacity of different networks (the default value was 1Mb/s). For pathChirp, the decrease factor is 1.5, busy period is 5 seconds, UDP payload size is 900 octets, test duration is 45 seconds and spread factor is 1.2. The duration and spread factor were carefully selected to obtain the stable available bandwidth value with 100 iterations. The minimum and maximum transmission rates are set to 10Mb/s and 200Mb/s for 100M Ethernet, 1Mb/s and 100Mb/s for WLAN, and 1Mb/s and 50Mb/s for EVDO. The values of the parameters for pathChirp...
were selected from a number of different parameter values of the parameters such that the selected ones provide better accuracy than others. SB uses the following default parameter values for all network types: $L = 576$, $N_{\text{min}} = 100000$, $N_{\text{max}} = 1000000$, and $w = 0.05$. The test packet transmission rate $R_t$ is unspecified. The statistic results for all three methods and the analytic results for SB for the forward and backward path measurement are compiled in Table 4. Since Iperf and pathChirp do not provide roundtrip measurement results, SB roundtrip results are listed separately in Table 5.

SB shows much less measurement latency (ML) regardless of packet loss (PL) than other methods for all network types. SB also maintains less number of octets generated (NO) than other methods for all network types, except for EVDO where NO of SB is greater than that of Iperf due to unspecified $R_t$ and relatively lower link capacity of EVDO. In order to further reduce NO, we recommend specifying $R_t$ to be closer to the link capacity if the capacity is known. The number of octets generated for Iperf depends on the value provided for the “bandwidth” parameter, especially for the backward path, since the target node is generating traffic to Ethernet. On the other hand, pathChirp NO for all network types are consistent under the same measurement latency. Next we will discuss the accuracy for different networks.

For Ethernet, Iperf and SB calculated the available bandwidth close to the actual network capacity for an 100Mb/s Ethernet, which is slightly less than the theoretical link capacity [6], where Iperf and SB packet sizes are 1470 bytes and 576, respectively. PathChirp apparently overestimated the forward path since the value is greater than actual Ethernet capacity. TCP throughput over the WLAN in our test bed, which is less than the available bandwidth. Iperf underestimated the available bandwidth for the backward path due to the significant packet loss over the wireless network. The SB and PathChirp underestimated the link bandwidth due to the fact that 802.11 link is a half duplex shared media. The test receipt packets interfere with the transmission of test packets. By reducing the amount of acknowledgement, SB with SACK of every 50 test packets provides the closest result.

For SB analysis, we calculate $\widehat{N}$ and $\widehat{D}$ by applying $R_t = B$, where $B$ is the bandwidth measured from SB, to the equations listed in Table 3. The value of $r_p$ is 1ms for Ethernet, 3ms for WLAN, and 85ms for EVDO. SB with unspecified $R_t$ generates more octets than $\widehat{N}$ since the sender generates test packets at its maximum processing speed. As a result, the packet loss ratio is higher when measured over the lower-speed link (i.e., EVDO) while the measurement latency

### Table 4: Measurement results for forward and backward path

<table>
<thead>
<tr>
<th>Network</th>
<th>Methods</th>
<th>Forward Path</th>
<th>Backward Path</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ML(ms)</td>
<td>BW(Kb/s)</td>
<td>VAR</td>
</tr>
<tr>
<td>Ethernet</td>
<td>IP(tx=100M)</td>
<td>1,007.0</td>
<td>97,753.09</td>
</tr>
<tr>
<td></td>
<td>PathChirp</td>
<td>45,000.0</td>
<td>103,283.28</td>
</tr>
<tr>
<td></td>
<td>SB(ack10)</td>
<td>59.21</td>
<td>93,397.07</td>
</tr>
<tr>
<td></td>
<td>SB</td>
<td>58.60</td>
<td>92,956.79</td>
</tr>
<tr>
<td></td>
<td>SB(analysis)</td>
<td>60.44</td>
<td></td>
</tr>
<tr>
<td>WLAN</td>
<td>IP(tx=20M)</td>
<td>1,142.30</td>
<td>9,608.60</td>
</tr>
<tr>
<td></td>
<td>IP(tx=50M)</td>
<td>1,131.00</td>
<td>10,219.31</td>
</tr>
<tr>
<td></td>
<td>PathChirp</td>
<td>45,000.0</td>
<td>12,474.18</td>
</tr>
<tr>
<td></td>
<td>SB(ack50)</td>
<td>108.21</td>
<td>11,034.72</td>
</tr>
<tr>
<td></td>
<td>SB(ack10)</td>
<td>186.50</td>
<td>5,814.42</td>
</tr>
<tr>
<td></td>
<td>SB</td>
<td>293.34</td>
<td>3,261.20</td>
</tr>
<tr>
<td></td>
<td>SB(analysis)</td>
<td>294.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PathChirp</td>
<td>45,000.0</td>
<td>444.89</td>
</tr>
<tr>
<td></td>
<td>SB(ack10)</td>
<td>478.00</td>
<td>86.51</td>
</tr>
<tr>
<td></td>
<td>SB</td>
<td>455.50</td>
<td>91.86</td>
</tr>
<tr>
<td></td>
<td>SB(analysis)</td>
<td>8,676.1</td>
<td></td>
</tr>
</tbody>
</table>

For Ethernet, Iperf and SB calculated the available bandwidth close to the actual network capacity for an 100Mb/s Ethernet, which is slightly less than the theoretical link capacity [6], where Iperf and SB packet sizes are 1470 bytes and 576, respectively. PathChirp apparently overestimated the forward path since the value is greater than actual Ethernet capacity.

### Table 5: Measurement results for roundtrip path

<table>
<thead>
<tr>
<th>Network</th>
<th>Meth</th>
<th>Roundtrip Path</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>ML(ms)</td>
</tr>
<tr>
<td>Eth.</td>
<td>SB</td>
<td>9.05</td>
</tr>
<tr>
<td></td>
<td>SB(a)</td>
<td>59.23</td>
</tr>
<tr>
<td>WLAN</td>
<td>SB</td>
<td>341.04</td>
</tr>
<tr>
<td></td>
<td>SB(a)</td>
<td>301.81</td>
</tr>
<tr>
<td>EVDO</td>
<td>SB</td>
<td>1,086.91</td>
</tr>
<tr>
<td></td>
<td>SB(a)</td>
<td>5,331.47</td>
</tr>
</tbody>
</table>

(Weth: Methods; a: analysis result)

WLAN available bandwidth depends on the wireless characteristics such as SNR and interference, which can vary throughout the time of the day. Average FTP traffic over the WLAN in our test-bed is 8,660.84Kb/s for forward path and about 9,292.37Kb/s for backward path over WLAN. This is the TCP throughput over the WLAN in our test bed, which is less than the available bandwidth. Iperf underestimated the available bandwidth for the backward path due to the significant packet loss over the wireless network. The SB and PathChirp with SACK underestimated the link bandwidth due to the fact that 802.11 link is a half duplex shared media. The test receipt packets interfere with the transmission of test packets. By reducing the amount of acknowledgement, SB with SACK of every 50 test packets provides the closest result.

For SB analysis, we calculate $\widehat{N}$ and $\widehat{D}$ by applying $R_t = B_t$, where $B$ is the bandwidth measured from SB, to the equations listed in Table 3. The value of $r_p$ is 1ms for Ethernet, 3ms for WLAN, and 85ms for EVDO. SB with unspecified $R_t$ generates more octets than $\widehat{N}$ since the sender generates test packets at its maximum processing speed. As a result, the packet loss ratio is higher when measured over the lower-speed link (i.e., EVDO) while the measurement latency
of SB forward test is smaller than $D$. Because the sender is generating test packets at a higher speed than the outgoing EVDO link capacity and a small socket send-buffer size (i.e., 8192 octets) is used, test packets are discarded at the sender before they are transmitted out to the network. The use of a small socket send-buffer size has the same effect as specifying a smaller $R$, and hence reducing the measurement latency.

When the packet loss ratio is small, the analytic results for forward and backward path tests are close to the measured results. For roundtrip results, the measured latency values deviate from the analytic results even for lower packet loss ratios. This is because in the SB roundtrip test implementation is optimized in that the source node immediately stops the test without sending Finish message (hence the measured latency does not include $w$ to send Finish message) if the algorithm stops transmitting Echo-Reply message before $N_{r}$ reaches $N_{\text{max}}$. This is why the measured ML for Ethernet roundtrip test is smaller than the analytic result. Nevertheless, the measured ML for WLAN roundtrip test is larger than the analytic result. This is probably due to forming two tight links in different directions as oppose to assumption (1).

Figure 6 shows the relationship between the measured available bandwidth and packet loss rate for Iperf and SB on the WLAN backward path. We observe a trend of higher packet loss introduces a lower measured bandwidth for Iperf, whereas SB available bandwidth results are independent of packet loss.

![Figure 6: WiFi backward path bandwidth v.s. loss rate](image)

**V. CONCLUSION AND FUTURE WORK**

We presented a new UDP-based available bandwidth measurement method called Single Burst. By comparing the SB measurement results with Iperf and pathChirp over Ethernet, WLAN and EVDO networks, we concluded that SB has short measurement latency, provides an accurate result when there is no cross traffic regardless of high packet loss rate, requires no parameter tuning for measuring over different types of network, and generates non-excessive test traffic. Though the EVDO measurement is tested in conjunction with other Internet traffic, we would like to further evaluate the SB behavior for Ethernet and WLAN with cross traffic as well as high RTT, for several different queuing disciplines including FCFS and SPQ.

**REFERENCES**


http://iperf.sourceforge.net/


**APPENDIX**

**A. Analysis of Forward and Backward Path Available Bandwidth Tests**

The sequences for forward and backward path tests can be considered identical if, in the backward path test sequence in Figure 4, the source and target nodes are swapped, the Start message is moved below the Finish message, and the Start message is replaced with a Report message. This means that the same set of upper bounds is derived for forward and backward tests. Therefore, we focus on analyzing forward path test. Let $t_{p}$ denotes the length of Data-ACK message including UDP and IP headers (i.e., $L_{d}=36$). Since Data-ACK messages do not contain a payload, $N_{p}=N_{d}L_{d}/L$. Therefore,

Equation 4 $\quad N = N_{d}(1 + L_{d}/L) \leq \max \{\min (R_{p}T_{p}/8L + L, N_{\text{max}}; N_{\text{max}} L)\} + 36/L$.

Let $T_{p}$, $T_{s}$ and $T_{f}$ denote the message delivery delay for the first Data and its Data-Ack, and the Report message, respectively. Let $L_{f}$ and $L_{d}$ denote the length of Finish and Report message including UDP and IP headers, respectively. Since $T_{f} = T_{p} + T_{s}$, the following relationship holds.

Equation 5 $\quad D = T_{p} + L_{d} + 8L_{d}/B + T_{s} + w = T_{p} + w + 8(N_{d} - L)/B + 8L_{d}/B + T_{s} - t_{p} \leq T_{p} + w + 8[\min (\{R_{p}T_{p}/8L + L, N_{\text{max}}\}; N_{\text{max}} L) - L]/B + 8L_{d}/B + T_{s} - t_{p} = 2T_{p} + w + 8[\min (\{R_{p}T_{p}/8L + L, N_{\text{max}}\}; N_{\text{max}} L) - L]/B$.

**B. Analysis of Roundtrip Path Available Bandwidth Test**

Since Echo and Echo Reply message have the same message length, $N_{e} = N_{f}$. Therefore,

Equation 6 $\quad N \leq 2 \max \{\min (R_{p}T_{p}/8L + 1; L, N_{\text{max}}; N_{\text{max}}\}) = 2 \max \{\min (R_{p}T_{p}/8L + L, N_{\text{max}}; N_{\text{max}}\})$.

Equation 7 $\quad D = T_{p} + L_{d} + w \leq T_{p} + w + 8[\min (\{R_{p}T_{p}/8L + L, N_{\text{max}}\}; N_{\text{max}} L) - L]/B$. 

978-1-4244-4148-8/09/$25.00$ ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.