

Locating Performance Problems in Home Networks:

1 Proposal Summary:

Internet connectivity for a typical consumer is achieved through the linkage of a wireless basic service set (BSS) link to a local Internet Service Provider (ISP) access link, which subsequently connects to a wide area network (WAN). The connection point between the BSS and the ISP access links is completed through a wireless router, also referred to as an Access Point (AP)¹. Performance bottlenecks, either within the BSS or access links, can degrade a consumer's internet experience.

A software tool has been developed by a research team at Georgia Tech to evaluate and determine if performance bottlenecks lie within the BSS or access links². Special software installed on the AP will passively and unobtrusively process characteristics of both links. This software will collect and process packet timing data pertaining to both links in addition to bitrate and re-transmission data on the wireless link. The processed data (meta-data) is then anonymized and relayed to a server³. Software on the server will evaluate the meta-data to determine if and in what link a bottleneck resides. The system wide conceptual view is shown in Figure 1.

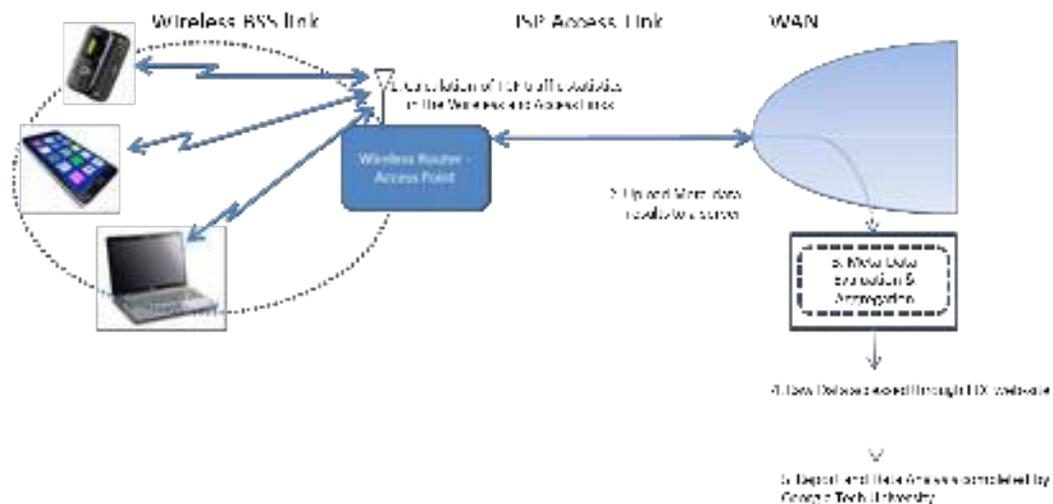


Figure 1

By utilizing equipment that has been deployed in the Measuring Broadband America (MBA) program, the goal of this work is to gain a better understanding in where internet bottlenecks exist.

2 Methodology:

A. Creation of Meta Data on the AP

Software, residing on the AP router, will passively and unobtrusively process the following characteristics on both links:

- (a) Access Link/WAN: TCP *packet timing* characteristics⁴ and
- (b) Wireless BSS Link: TCP and Data Link *bitrate, frame re-transmit, and round trip timing (RTT)* characteristics⁵.

The software is designed to capture data every 5 minutes for a 15 second period. In the event that traffic flow during a capture period drops below 100 packets per second (pkts/sec), the software disregards (and does not collect) data during that portion of the period. However, if the traffic flow rises above 100 pkts/sec later in the capture period, the software will then re-start collecting data.

Access Link Packet Timing Meta-Data:

The AP software collects timestamps from incoming TCP traffic packets with *tcpdump*⁶. After the completion of a capture interval, the AP software computes a coefficient of variation of packet inter-arrival time, c_v . c_v is calculated from the observed differences between successive timestamps of the incoming traffic. It is computed as the standard deviation divided by the mean of the packet inter-arrival time and is stored on the AP as meta-data.

Wireless Link Bitrate, Frame Re-Transmit, and RTT timing Meta-Data:

Concurrently to collecting data for the access link, the AP software will use *tcpdump* to capture wireless link information, which, for each frame includes: the bitrate used, whether the frame was retransmitted (ρ) and the associated packet time stamps for source and destination MAC addresses of connected stations.

The bitrate for the payload in each 802.11 Physical Sublayer frame is given in the Signaling Field of the Physical Layer Convergence Protocol (PLCP) header. Since this value can vary for each frame and in each direction, the bitrate value used in the analysis is the lower of the downstream and upstream average observed.

The frame re-transmission rate is derived from bit information given in each 802.11 MAC Sublayer header. In this MAC header, the field Frame Control indicates only if the frame has been re-transmitted. Software on the AP adds up the number of frame re-transmissions per second to compute the frame re-transmission rate.

Using the output of the *tcpdump*, *tcptrace*⁷ produces information regarding RTTs (τ) between the AP and end-hosts for individual traffic flows. Since several traffic flows can originate or end at one device, the RTT is computed as an average across all flows per device.

Once the meta-data of the access and wireless links are determined for each capture period, the AP anonymizes all IP address and MAC addresses completely by using SHA-256 in addition to a per-router secret salt. The router discards all private information and uploads the meta-data to a server. After the upload is completed the router deletes the local copy of data. Figure 2 depicts a flow chart that the AP software executes when creating meta-data.

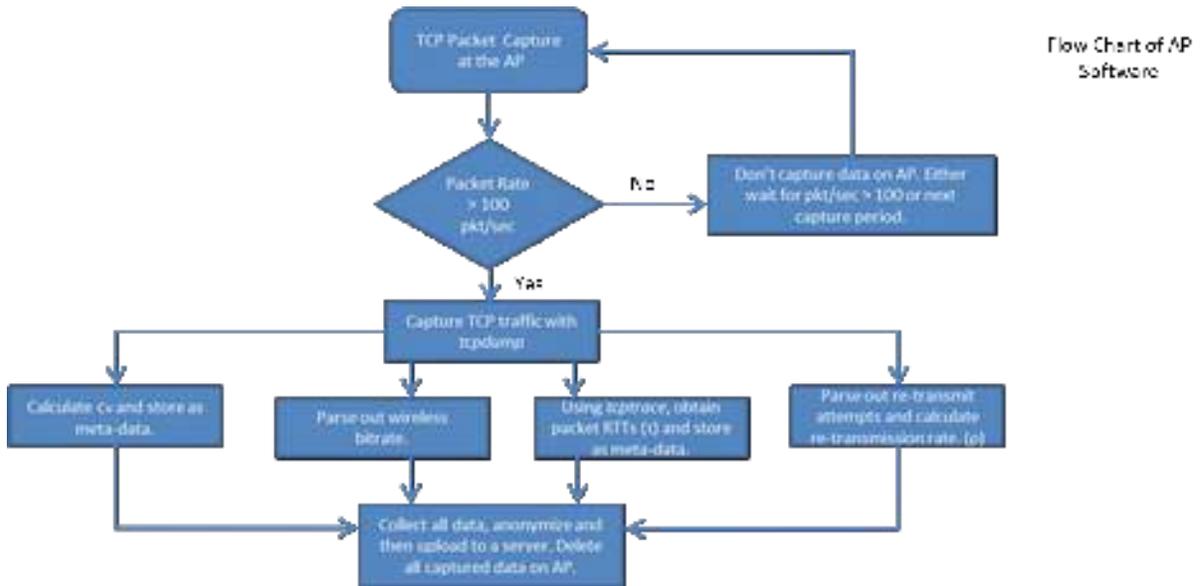


Figure 2

The meta-data is stored on a database within the server, where the diagnosis and longitudinal analysis portions of the study reside.

B. Analysis of Meta Data on the server

Before the server can diagnose the uploaded meta-data, it has to perform the tasks of normalizing the bitrate (μ) for each device connected to the wireless network. Once completed, the server is ready to evaluate the meta-data.

For the server to determine where a bottleneck occurs within the internet links, the server compares uploaded meta-data to predetermined threshold values. When comparing the meta-data to a threshold value, an algorithm will determine with a specified degree of confidence that a bottleneck does or does not exist within a specified link. The crux of this algorithm is based on the setting of threshold values. Threshold values are defined as the ratio of the conditional probability of an event happening to the conditional probability of the event not happening. The range of threshold values for the meta-data parameters are:

$$\begin{aligned} 0 < c_v < 1 \\ 0 < \mu < 1 \\ 0 < \rho < 1 \\ 0 < \tau \end{aligned}$$

In an ideal world, the observed values for c_v in the access and wireless links would be completely separated. Figure 3 represents this ideal world.

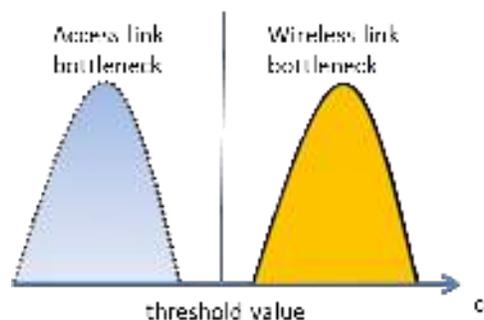


Figure 3

Unfortunately, this separation is not so clean and in fact observed c_v values for both links can overlap, as shown in Figure 4.

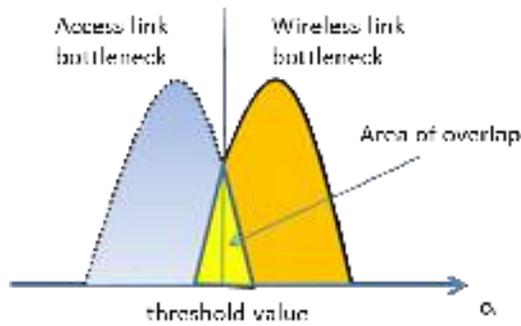


Figure 4

Depending on the amount of overlap there exists a possibility of observing a c_v above the threshold value and attributing it to a wireless link bottleneck when in fact it occurred due to an access link bottleneck. Similarly there also exists a possibility of observing a c_v below the threshold value and attributing it to an access link/WAN bottleneck when in fact it occurred due to a wireless link bottleneck.

Through empirical data, the algorithm will set a threshold value to maximize the observation of true bottleneck detections while minimizing the observation of false positive bottleneck detections.

A flow chart of the server bottleneck decision algorithm is shown in Figure 5.

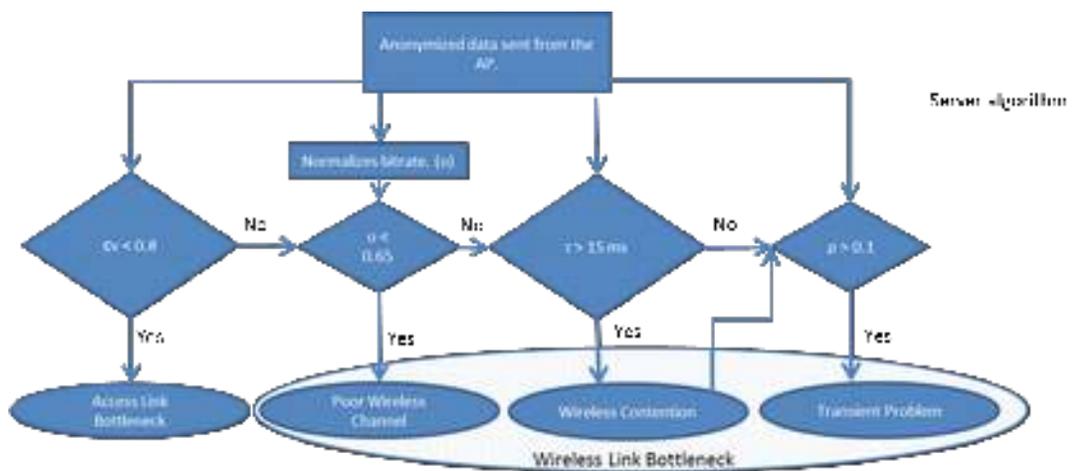
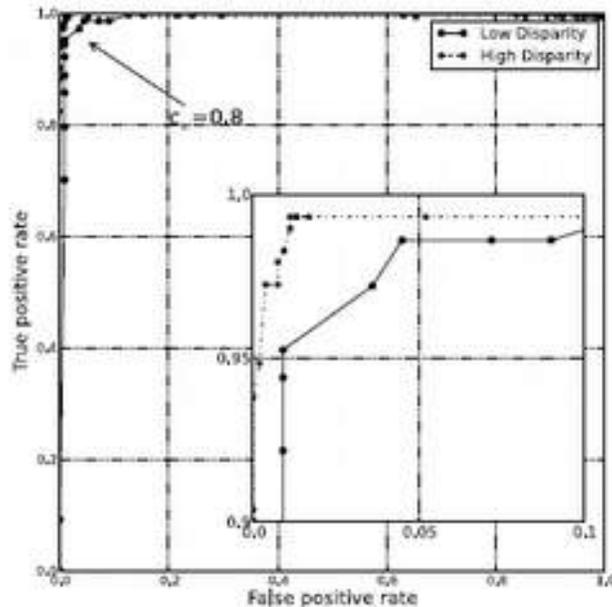


Figure 5

To determine if the access link contains the bottleneck, the algorithm compares c_v to a threshold value of 0.8. If c_v lies below the threshold then the access link is deemed the bottleneck, whereas if the value lies above threshold then the access link is not the bottleneck. Empirically, it has been found that the threshold value of 0.8 yields a nearly 95% detection rate for only a 2% false positive rate as Figure 6 shows.

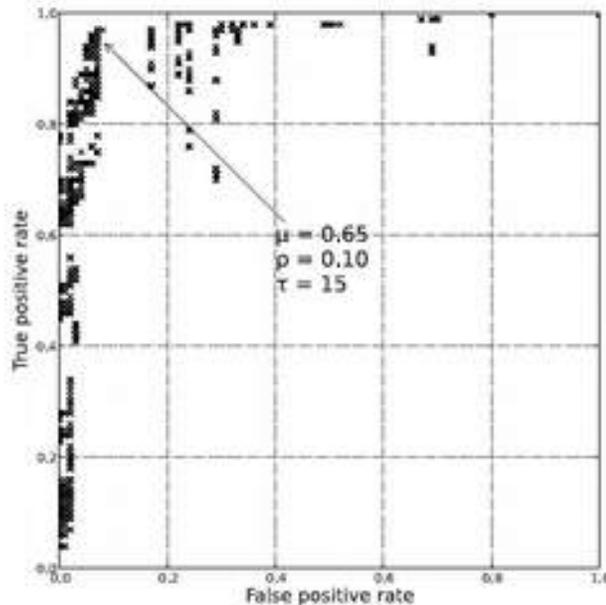


Threshold value for c_v
Figure 6

In the Figure, “Low Disparity” refers to the condition in which the access link throughput is close to the throughput of the wireless link. Likewise, “High Disparity” refers to the condition when either the access link throughput is much higher than the wireless link throughput or vice-versa.

To determine if the wireless link contains the bottleneck, the algorithm compares the meta-data of each parameter to its associated threshold value. Empirically, setting the threshold values of μ , ρ , and τ , to 0.65, 0.10, and 15, respectively, yields a 97% detection rate for

only a 7% false positive rate. Figure 7 depicts the False Positive Rate versus the True Positive Rate with the set thresholds as described above.



Threshold values for μ , ρ , and τ
Figure 7

3. Limitations:

These are enumerated below:

- (1) The software can only distinguish traffic flow characteristics across an AP – i.e. it can only evaluate if a bottleneck occurs in either the access or BSS links and it makes no attempts to separate ISP and WAN performances. Furthermore, for the software to make a decision on where a bottleneck exists, it requires a reasonable amount of traffic flow across the AP.
- (2) Since these threshold values are derived from probability calculations, the decision algorithm attempts to maximize a correct decision while minimizing the potential for a false positive. Equal, or close to equal, data rates in both links will lead to the “Low Disparity” graph as detailed in Figure 6. In

this situation, the number of false positive detection increases.

- (3) Finally, the AP needs to be configured as a router in order for the software to accurately record traffic meta-data characteristics. Configuring the AP as a bridge would indicate an AP that is not the end-host (gateway) of the BSS network. This study is focused on the overall traffic characteristics between an end-host and client. Measuring traffic characteristics at intermediate points might not accurately represent the end-host to client connectivity.

4. Data Analytics and Publishing:

Raw data in the form of the meta-data statistics c_v , μ , ρ , and τ as well as the conclusion about the existence and form of pathology generated from the passive tests will be periodically transferred to the FCC web site. This data will be openly available to the public. Data analytics and report publishing of this study will be conducted through an academic team at Georgia Tech University.

- 1 The AP hardware, or "Whiteboxes", used in this study is provided by SamKnows. These Whiteboxes are self-installed by the consumers.
- 2 This software is designed and implemented to run on OpenWrt file-system. More information about OpenWrt can be found at: www.openwrt.org
- 3 The server is part of the cloud storage system managed by SamKnows.
- 4 Coefficient of variation will be calculated from Layer 3 timing information from the OSI model.
- 5 Bitrate and Frame re-transmit are informational fields in Layer 2 of the OSI model. Round trip time will be calculated from Layer 3 timing information from the OSI model.
- 6 www.tcpdump.org
- 7 An open source software tool. More information can be found at: www.tcptrace.org