Home WiFi Impairments Detector

Israel Márquez Salinas
Université Pierre & Marie Curie
s.marquez.israel@gmail.com

ABSTRACT

Home WLANs have become an essential element in households nowadays. The preferred method to access Internet from home is WiFi. Home WLANs have brought their benefits and challenges into the home. The variety and complexity of WiFi and non-WiFi devices make Home WLANs keen to experience WiFi impairments. Identifying these impairments can be challenging, even for Wireless experts. To approach this challenge we have begun to develop a tool to identify WiFi issues in Home WLANs. In this paper we present the initial stages to develop the basis of this tool. We have conducted experiments triggering Wireless and non-wireless issues in a testbed. During these experiment sessions we have collected metrics from different components in the setup. Metrics have been collected using active and passive measurement techniques, a description of these two techniques is covered in section 2.1. Finally we consolidate and correlate these metrics to identify when a WiFi issue is happening.

1. INTRODUCTION

Networks today have evolved significantly, one of the most tangible examples of this evolution are Wireless Networks. The most common way to access Internet from home are home WLANs, usually referred as home WiFi. The variety of services and devices using the home WiFi to access Internet is vast. It is common today for a home user to stream a movie on his laptop while connected to the home WiFi. In the ideal case scenario the experience is enjoyable, video team plays smoothly. In many cases, when the movie streaming is degraded, the experience is frustrating. One of the potential causes of poor streaming experience is the home WiFi. In fact, previous works [12] have identified home WLANs as the bottleneck along the service path. The cause of poor Home WLAN experience can be varied, as described in previous work [5], channel congestion, poor client or AP placement and interference are the most common causes. Other works [9] have analyzed the impact of Home WLAN on latency along a network path. They have identified that WiFi hops latency can contribute up to 60% of the overall round trip time along the service path. On top of these technical causes, a business risk arises. A risk between home users, ISPs and content providers. As described in previous works [1] the frustration is not only experienced by the users but also by users' ISPs who are often held responsible for poor Internet experience. This problem might seem small, nevertheless it can escalate until a point in which content providers lose their subscribers. Home users, in the search of a solution can switch between ISPs, if complication persists they can even switch content providers. In this context ISPs and content providers have little to none impact on one of the most common root causes of the degraded experience, the home WLAN. Under this light lies our motivation to develop a tool to identify Wireless impairments in Home WLANs. The description of the initial stages of this tool are presented along this paper. Identifying where the root cause is within the Home WLAN is challenging due to multiple factors. To begin with, Wireless nature is unreliable as it uses an open and shared medium, shared among WiFi and non-WiFi devices. Another factor is to choose the most suitable measurement technique to find where the problem is. At the time of this paper and to the best of our knowledge, most research works have mainly implemented passive techniques [3] [2]. A couple others have relied upon active techniques [4]. Depending on the type of measurement technique chosen challenges can be presented. Passive techniques face the challenge of requiring access to the device collecting the metrics. With active techniques the complication is tied to overhead caused by the measurement tool. In other words, with active techniques the very same measurement instrument can bias the measured metrics. Our tool implements a mixture of both to take strong points of both and leverage the weakness with each other's strong points. Further description of these techniques along with related work associated to home WLAN study will be covered in section 2. The instrumentation details of our tool are developed in section 3. The mechanisms and techniques to evaluate our method to identify impairments in Home WLANs is explained in section 4. Finally findings of our work are consolidated

in section 5.

2. BACKGROUND AND RELATED WORK

The challenge to identify issues in the Home WiFi has been approached before. To address this challenge the research community has relied on two measurement techniques, active and passive. While most of previous works have opted for passive techniques [3] [2], others, have worked with active ones [4]. In the work of Joumblatt, Diana, et al. [3] they have opted for passive techniques to be able to extract fine-grained data from packet captures. In their experiments context active techniques were not chosen as they might not reflect application performance properly. Application performance was a key element of their research. They overcame privacy concerns associated by anomymizing the collected data. The work of Neves Da Hora, et al [2] also chose passive techniques. In their research context, active techniques might have led to user traffic disruption and battery drain of devices under study. Within the context of passive metrics, they excluded per packet analysis as it can result in overhead during high network utilization periods. Their work mostly relied on standard metrics passively collected from APs. Active techniques were implemented in the work of Kanuparthy, Partha, et al [4]. Their work rely on user-level probing. They propose a metric called one-way-delay OWD or wireless access delay. The OWD reflects the delays a packet faces while going through a 802.11 link. They have chosen active measurement to achieve software and hardware agnostic mechanisms. They pursue agnostic mechanisms to facilitate the deployment of the tool at a large scale. The common ground among the works mentioned before and our work is tool's usability and scalability. We strive for a tool to be deployed at a large scale with minimal modifications to the Home WiFi setup. The area in which we differ with previous works is the implementation of an active measurement technique with a specific probing rate. The probing rate we have chosen will get a sense of network status without adding significant overhead to it. We describe how we chose the probing rate in Section 3.

2.1 Wireless Monitoring Metrics

Active and passive techniques have their own strong points and areas for improvement. In the following lines we outline the main characteristics of each one of them and what can be considered their strengths and weaknesses. Important to mention, we do not dare to tell a specific technique is better than the other. Each of the techniques will be best-suited depending on the goal and context of the experiment.

Active

Active measurement techniques are mainly characterized by its ability to capture the state of the network in almost real-time. In other words, active measurement can help to identify a condition when is present in the network. This characteristic is different from passive measurements which can be considered historical. Active measurements are also characterized by the use of probes. Probes are packets "injected" in the network to measure its status. For example, ping relies on ICMP requests and replies to compute the Round-Trip Time. For Ping, the probes are the IMCP requests and replies. It is important to pay attention to the probe size and probing rate. Probes can add overhead to the network if their size is large compared to the capacity of the path or if the rate is high. If probing causes overhead it will not only might disrupt user traffic but can also lead to biased measurement results. In the following bullet points we outline the strengths and weaknesses of active measurement techniques.

Strengths

- Full ownership of the network is not required.
- They do not require large space to store data collected as generally, probe packets are small.
- Privacy concerns are minimal as probe packet used to measure are made of random data which has no sensitive information.
- Useful to get the state of the network in almost real-time.

Weaknesses

- Overhead might occur if probe size and rate are chosen without due diligence of network conditions.
- Biased results can be obtained if probing causes overhead in the network.
- They can only capture an instant of the network condition. If problem to be characterized is extended in time, active measurement might not measure it accurately.

Under the scope of active measurement techniques, the following are the metrics to be actively collected for our work.

Active Metrics

- **Round Trip Time**
  - This metric takes into account the time it takes for a probe to leave the source, reach the destination and come back to the source. From the RTT we have obtained statistic, i.e. Average, Std. Deviation, Max and Min.

- **Throughput**
  - The amount of data sent or received from or by a station within a time window.

Passive

Passive measurement techniques rely on a "listen and sit" approach. The instrument conducting passive measurements in the network sits in a specific location along the path and records the metrics of interest. The instrument can be a component of the network itself, for example a router. It can also be device devoted to measure, such as a Wireless sniffer. An important difference between active and passive techniques, is that passive tend to be historical whereas active are real-time oriented. In an historical sense, passive measurements are more reliable to characterize a network problem which covers an extended time-frame. Active measurements are best-suited to pinpoint a problem in the instant it happens, nevertheless they lack accuracy to characterize problems covering an extended time-frame. Another difference between active and passive measurements is that the latter do not trigger probes. Overhead due to probe packets is not present in passive measurements. However, computational and storage resources in the passive measuring device are important factors to consider. The device might require to have enough space to store the data being collected. In a similar way, the computational power of the device can be required to be high depending on the speed of the link being measured. A Gigabit link in a Core Router will handle significantly more data than an 100Mbps Ethernet link in an access switch. Outlined in the following list a high level summary of the strengths and weaknesses of passive measurement techniques.

Strengths

- No extra traffic is generated to collect metrics, risk of causing overhead is minimized.
- They are best-suited to accurately characterize network problems covering an extended time frame.
- In general, they are able to collect large datasets leading to fine-grained data. Ultimately leading to increased network complications diagnosis accuracy.

Weaknesses

- Large storage capacity can be required to store collected data. Not all measuring devices have large storage capacity, i.e. Access Points.
- Access to equipment working as passive measurement device is required. This is not possible for most users at multiple devices along an Internet path.
- High computational power on the measuring device can be required depending on the link being monitored and data granularity pursued. Not all devices can provide high computational power, i.e. Access Points.
- They are reactive, findings on the network problem can be obtained after collected data has been analyzed. The majority of wireless interference issues are known to be short, 1 - 7 min [8]. By the time the data has been analyzed the wireless interference issue might be already over.

Passive Metrics

- **RSSI - Received Signal Strength Indicator**
  - The power at which the signal is being received by the device. Depending on the type of traffic, specific RSSI thresholds are often defined to set an acceptable RSSI level. For example, for VoIP the min RSSI value for an acceptable VoIP call is -67 dBm [6].
- **PHY Tx Rate**
  - The rate at which without medium access control, error correction or scheduling events the device is expected to operate with.
- **Noise**
  - The noise perceived in the Wireless environment, high noise levels can degrade Wireless link quality.
- **Throughput - Driver Logs**
  - For this metric, we extract the throughput perceived by the Wireless driver debug logs. It depicts the effective amount of data that can be exchanged.
- **Frame Delivery Ratio**
  - Frame Delivery Ratio depicts the ratio between packets successfully received and total packet sent. The FDR metric can assist to get a sense of link quality. If FDR ratio is high then, the quality of the link can be perceived as good.

2.2 Vantage Points

In our work we have collected the metrics described in section 2.1, from multiple devices, vantage points. The device from where we have collected the metrics is key to identify which device is perceiving a particular Home WiFi issue. The metrics values can differ depending on the vantage point, the difference can be caused by device placement, OS, resources, driver and many more.

- **Throughput - Active** - We have actively measured throughput from the wired client. As the goal is to test the link between Wireless client towards the Wired Client and going through the AP, we have collected the data at the wired client. Previous works have identified that even with similar Wireless conditions devices can experience different throughput and bitrates [11]. We use iPerf as the tool to collect this metric. Further details on iPerf setup to collect this metric are described in section 3.

- **PHY Tx Rate** - We have extracted this metric from WiFi logs at AP and Wireless client. The goal is to identify at which rate was the last frame prior to collecting the log sent. We have also added a wireless sniffer to collect packet captures of wireless traffic between wireless client and AP.

- **RSSI** - We extract logs from the Wireless client and the AP to obtain RSSI data from each of them. Different logs have been collected at these two vantage points to validate its accuracy.

- **Noise** - A factor contributing to Wireless degradation is Noise, it is the Wireless interference coming from non-Wi-Fi sources. This can be caused by Microwave ovens, cordless phones and similar devices which "do not speak Wi-Fi language". Noise will be measured at both ends, wireless client and AP. We strive to identify which one experiences higher noise levels to pinpoint where the complication might be located.

- **Frame Delivery Ratio - FDR** - To compute the FDR we have fetched driver debug logs from the AP and the Wireless client. The FDR can assist to identify which component, AP or Wireless device is experiencing a poor Wireless condition. For example when experiencing congestion, the AP can have a lower FDR than the AP as the AP location do not experience high channel utilization.

- **Throughput - Driver Logs** - From driver debug logs we collected Throughput at the AP and Wireless client. The goal is to validate the closeness between the perceived passive throughput between AP and wireless client.

- **RTT** - At the wired client we issue pings towards the wireless client. We log the ping output at the wired client to compute statistics from the RTT. Statistics collected from the RTT are minimum, average, maximum, standard deviation and losses.

3. WIRELESS BOTTLENECK CLASSIFIER

We strive to identify when a Home WiFi issues is present. To achieve our goal we have ran experiments in which we caused WiFi and non-WiFi issues in a test bed. During the experiment sessions we have collected active and passive metrics using a diverse set of tools. Most of the tools we have worked with are out-of-the-box tools, such as iPerf, tc and tcpdump. The active measurement tool we have used to collect RTTs is a custom implementation of Ping in GoLang. We refer to this tool as *GoPing*. We have customized GoPing to send ping trains and batches. These two features were key to find the probing rate to be used in later stages of our experiments.

3.1 Finding the probing rate

As described in section 2, finding the probing rate is important when working with active measurements. A high rate can cause overhead, whereas a low rate can fail to capture the status of the network. To approach this challenge we conducted experiments in our office lab. Our experiments consisted in sending a series of ping trains which included multiple pings inside each train. We ran the tests with different train inter-spacing values and with different amount of pings inside the trains. The first finding from our experiments was a delay in RTT due to power save mode in devices. The power save mode sends the NIC to sleep. We refer to this delay as the "sleeping NIC". We found that when the inter-train spacing is smaller or equal to 100 msec the power save mode delay is not present. Based on this finding we set our lower bound for inter-train spacing to 100 msec. We set our upper bound to 1000 msec as the RTT within a home WiFi single-hop network is expected to be only a few milliseconds. This observation is remarked in the work of Sundaresan et al. [12].

The second relevant finding is associated to the RTT value of each ping within a train. We found that even with inter-train spacing values above 100 msec it is possible to overcome sleeping NIC delay by considering the RTT value of the 3rd or greater pings within a train. We noticed that the RTT value for ping greater or equal to the 3rd ping in a train depicted similar RTT values as when the sleeping NIC delay was not present. After these observations we defined our baseline to be 100 msec inter-train spacing and 3 pings per train. Figure 1 illustrates the values for the average round trip time of the three pings in a train. The inter-ping spacing is equally distributed based on the number of pings in a train and the inter-train spacing. For example, the inter-ping spacing for a series of train spaced by 100 msec and with three pings inside each train is 33.33 msec.

Once our baseline was defined, we implemented similarity tests between our baseline results and samples derived from the baseline. We refer to our baseline as aggressive probing. To keep the samples to follow the same distribution as our baseline we implemented a

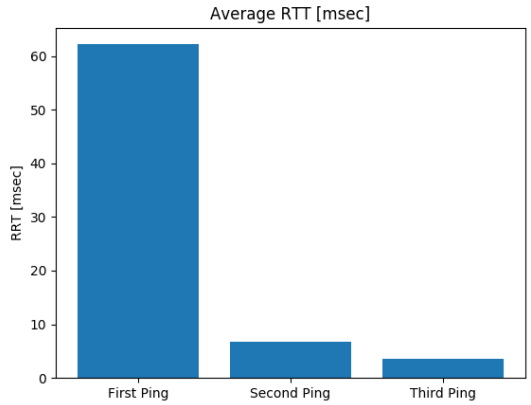

Figure 1: Average RTT for Three Ping Series

Poisson process to generate the inter-train space intervals. In other words, randomly sampling from a Poisson process will result in another Poisson-distributed process [10]. This feature has been included in our GoPing tool. From our baseline we sampled 10 - 90 % of our original baseline data points. We implemented a Bernoulli random sampling to extract the samples. Finally, we ran Two-sample Kolmogorov-Smirnov tests between our baseline and samples. From the results we noticed that the sample which delivers a similar ECDF to our base line is the one keeping 50% of the original baseline data points. Figure 2 illustrates both ECDFs.

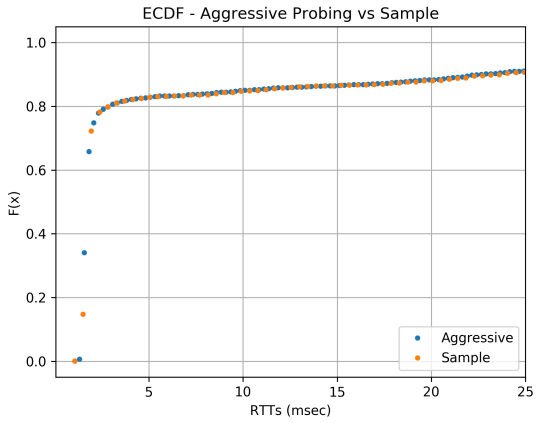

Figure 2: ECDF - Aggressive Probing vs Sample

From figure 2 the overlap between the sample keeping 50% of original data points and the original baseline can be seen. This was the sample in which the overlap began to occur, based on this result we obtained our probing rate. As the RTT ECDF of the sample with half of the original data is similar to the original baseline, we set our probing rate to be 200 msec. To validate our chosen probing rate still holds in our testbed we conducted tests. The tests consisted in sending as many batches as possible for 10 min at 100 and 200 msec probing rates. Additionally we varied the attenuation with values of 0, 15 and 30 dBm. Table 1 summarizes the values used for the test. The test sessions were conducted in the 2.4 GHz band using 802.11n WLAN with no authentication. Each of the experiments was conducted 5 times, in total we obtained 30 samples.

Attenuation	Probing Rate
0 dBm	100msec
0 dBm	200msec
15 dBm	100msec
15 dBm	200msec
30 dBm	100msec
30 dBm	200msec

Table 1: Attenuation and Probing Rate Validation Values

To validate the similarity between the probing rates in the testbed we validated similarity between the two probing rates. We compared the RTT ECDF of both rates. The expectation was for curves to be similar to each other. As expected, figure 3 illustrates the similarity between RTT ECDF of both probing rates.

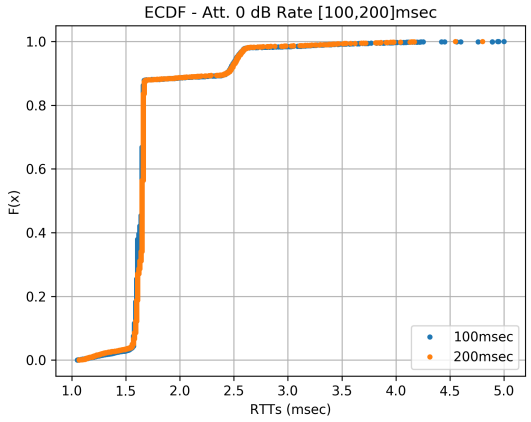

Figure 3: Att. 0 dBm - Rate 100,200 msec

The next expected behavior was for RTT to increase as the attenuation values increase. Figure 4 help us to validate the expected behavior. As we increase attenuation, RTT increases.

4. EVALUATION METHOD

In order to find our probing rate for run our experiments we worked with two testbeds we had access to. The first is a small setup in our office, the second is a testbed called Orbit Lab [7].

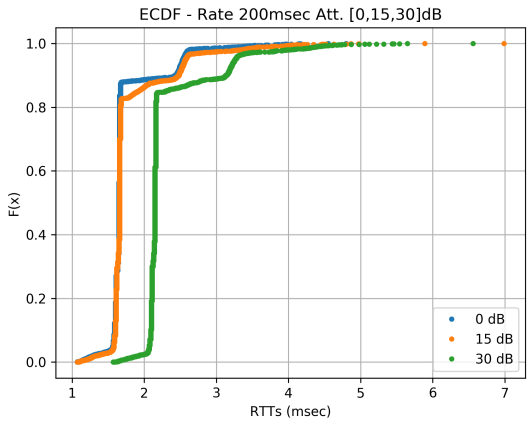

Figure 4: Rate 200 msec - Att. [0,15,30] dB

In-Office Lab

The In-office lab was mainly used to work on finding the probing rate at which the test in Orbit lab testbed were going to be conducted. The setup at our office is primarily composed by the following components.

1. Raspberry Pi 3 running Raspbian GNU/Linux 8 (Jessie)
2. Wireless Access Point TP-Link AC1750
3. Dell Laptop Inspiron running Ubuntu 16.04.4 LTS (Xenial Xerus)

The wireless card driver on the Dell Laptop supported 802.11 a/b/g/n/ac. The driver is *iwlwifi* version 4.4.0-130-generic and firmware=17.948900127.0. At the In-Office lab we setup our deployment as illustrated in figure 5.

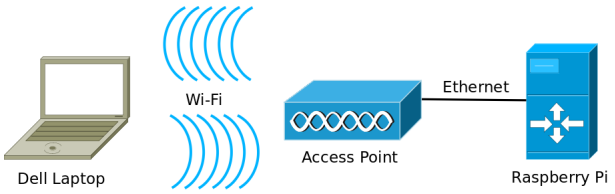

Figure 5: In-Office Lab Deployment

The Pi was the device from which the Pings were sent towards the Laptop. As illustrated in figure 5 the Pi and access point were connected via Ethernet. The laptop was connected with the AP via 802.11n. Throughout the tests at the In-lab office we switched between 2.4 and 5.0 GHz band depending on the goal of the experiment. The In-Office lab played a key role to test the features we included in our GoPing tool prior to running experiment at Orbit lab testbed.

Orbit Lab

The second testbed we worked with was Orbit Lab [7]. Orbit lab can be considered a large testbed in which different Wireless technologies can be tested. One of these Wireless technologies is 802.11. Within Orbit Lab we worked with Sandbox 4 (SB4) which includes features to vary the attenuation between the nodes in the Sandbox. The main components of SB4 we worked with are the followings.

1. SB4 has 9 nodes, each one of them runs Ubuntu 12.04
2. Attenuation Controller which makes possible to vary the attenuation between the nodes.

Each of the nodes has an Atheros Wireless card, the models are Atheros 5K and 9K. The nodes we worked with have Atheros chipsets which allowed us to collect detailed logs. The links between the nodes in SB4 can be set to attenuation values between 0-30 dBm from the attenuation controller. The topology of SB4 depends on the attenuation values for each link. For example a full-mesh topology is achieved when the attenuation value for all the links is set to 0 dBm. The feature of setting different attenuation values allowed us to test two scenarios associated to WiFi issues. The main deployment we worked with is a replica of the deployment we setup the In-office lab. The node working as AP was setup using the *hostapd* utility. The WLAN settings for the AP are summarized in table 2.

Setting	Value
802.11 Protocol	802.11n
Channel Bonding	No
Band	2.4 GHz
Security	Open

Table 2: WLAN Settings at AP Node

Experiments Setup

In order to trigger WiFi and non-WiFi issues we setup three main scenarios in Orbit SB4 testbed. Two of them were associated to WiFi issues, attenuation and congestion. The third one was an issue at the access link. Across the three scenarios logging, GoPing, Wireless capture and iPerf measurement are similar. Our experiment sessions lasted for 10 min. During the session we collected Wireless stats logs every 10 secs at the AP and Wireless client. We also setup a Wireless a Sniffer to obtain Over-the-Air packet captures. From the wired client we sent the pings towards the AP and log the stats from GoPing. For the throughput measurement with iPerf, we setup iPerf server at the wireless

node and the iPerf client at the wired node. iPerf was setup in TCP mode with 4 parallel TCP streams. We chose TCP as it is transport protocol most commonly used by services at Home WiFi networks. We decided to work with 4 parallel streams as within a Home WiFi the number of TCP in average is below 5 TCP streams [**We need a reference here**].

Attenuation

To trigger attenuation impairments in the testbed we varied the attenuation at the link between the wireless client and the access point. The deployment we worked with derived from the one illustrated in figure 5. The only additional component is the node working as a wireless sniffer. The attenuation issue was triggered in third and ninth minute of the 10 min session. The experiment was designed this way to set a comparison between non-issue and issue conditions. We setup 5 scenarios with an increase of 3 dBm per impairment interval. Table 3 breakdowns the scenarios and the attenuation levels for each impairment interval. Each one of the experiments was run 5 times.

Scenario	Attenuation Value [dBm]	
	1st Interval	2nd Interval
1	3	6
2	9	12
3	15	18
4	21	24
5	27	30

Table 3: Attenuation Scenarios and Values

Congestion

To trigger congestion in our testbed, we connected more wireless nodes to the same AP our main wireless client was connecting to. At the 4th and 8th minute we connected an additional wireless node to the AP. The additional wireless node client sent iPerf UDP traffic to the AP, hence the AP was running an iPerf server instance. We increased by one the wireless nodes connected to the AP by one per issue interval. Table 4 consolidates the scenarios and number of wireless nodes connected to the AP.

Access Link Limiting

The third scenario experimented in the testbed was a non-WiFi issue. The issue consisted in limiting the access link capacity at the Wired node. We achieved limiting at the wired node by using *tc* which is a traffic shaper. In a similar way to the attenuation scenario, we triggered the issue at the 3rd and 9th minute of the 10 min experiment window. Table 5 details the scenarios and values used for the Access Link Limiting scenario.

Scenario	Connected Wireless Nodes	
	1st Interval	2nd Interval
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

Table 4: Congestion Scenarios and Values

Scenario	Throughput [Mbps]	
	1st Interval	2nd Interval
1	100	90
2	80	70
3	60	50
4	40	30
5	20	10

Table 5: Access Link Limiting Scenarios and Values

- **Attenuation**
  - We have been using the embedded manger for attenuation in Orbit.
  - We can instrument attenuation values on the links connecting the nodes, in our case we vary the attenuation values between Wireless client and AP.
  - Attenuation controller allows to define values in the range from 0 - 30 dBm.
  - For our experiment we have been varying the values from 0 to 30 in steps of 3.
  - We vary the attenuation and record the RTTs for pings.
  - We have identified that after 27 dBm of attenuation is when we begin to see an increase in RTTs, each session last 10 min. Probe rate every 200msec.
  - At 30 dBm the connectivity between Wireless client and AP is lost.
  - For bandwidth test we have run iPerf and recorded the bandwidth obtained at the client side.
  - **With 5GHz we identified that after 6dBm the connectivity between client and AP is lost.**
  - We have setup 802.11n using 2.4GHz band to increase the range.
  - The goal is to run iPerf and identify at which attenuation levels does the bitrates drops, record the attenuation values to run ping tests.

- Once the attenuation values have been identified the next step is to run ping tests using the attenuation values found with iPerf test and record the average RTTs.

Main ideas for Orbit test bed description

1. Orbit is a testbed mostly devoted to Wireless experiments. (Mostly as they also have SDN sandboxes to test SDN technologies)
2. We have been using the Sandbox 4, SB4, which is devoted to Wi-Fi and Wi-Max Experiments.
3. SB4 is made of 9 nodes, each of them runs Linux based systems, Ubuntu 12.04 to be precise.
4. Our main setup is composed by three nodes. One node plays the role of the AP, another the role of Wireless client and the last one is a Wired client.
5. We are using 802.11n in 2.4GHz band to improve the reachability of the AP and the Wireless client.
6. The wired client is the source of the probes and iPerf server.
7. Wireless client plays the role of iPerf client.
8. We can include a diagram of the Orbit SB4 deployment and include the proper references.

Main ideas for the evaluation methods

- **Interference**
  - Currently looking for a way to create Noise in SB4.
  - Check if for a specific time they can setup a Microwave oven or similar.
- Congestion
  - For this experiment we will deploy a second wireless client connected to the same AP.
  - The 2nd Wireless client will send traffic to the iPerf server located in the wired client.
  - The original client will continue to send pings to the wired client.
  - We will record the results of RTT while other client is sending traffic to the iPerf Server.

We used three nodes with. Atheros 9k and 5k wireless cards.

We configure a node to work as a Wireless station, another as an AP and finally a third one as a wired client from where the pings were issued.

The third node working as a wired client plays a similar role as the Pi in our In-lab setup.

4.1 Setup of testbed

Here we explain how we ran the experiments.

4.2 RSSI in the wild

In order to collect realistic metric from can be considered a common value of RSSI in the wild we ran survey to collect this metric. We asked our colleagues in our office to run a script from which's output we can extract the RSSI value. We obtained 760 samples metrics coming from different environment contexts, mainly home and offices. We found RSSI average value in the wild to range between -60 and -60 [dBm]. Following picture depicts the histogram of RSSI obtained from the survey.

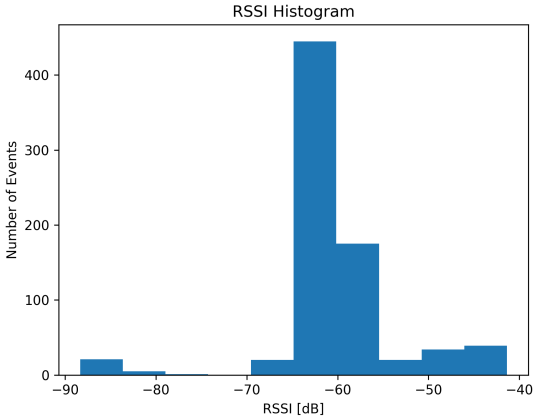

Figure 6: RSSI Survey Values Histogram

The main goal of this exercise is setup our testbed attenuation settings to trigger an RSSI value similar to the one found with the survey. In our testbed the attenuation values which lead to an RSSI value between the found range are 0, 3 and 6 [dBm] in the 2.4 GHz band.

We can include the accuracy of our methods depending on where are we setting our Vantage point.ex

In our lab we placed the laptop and the Pi close to each other, a distance smaller than 5 m. We connected to the 5GHz band under 802.11n protocol.

The first set of experiments consisted in progressively adding TCP sessions. The goal was to perceive how was RTT changed with more TCP sessions. We expected to see an increase as more TCP session were added.

Results matched our expectation and saw an increase in average RTT as more TCP session were added.

Include plot in which we have the CDF of RTTs vs TCP Streams

The next set of experiments were ran with the goal of finding a suitable probing rate. The ideal case is to probe frequent enough to have a "good" sense of the network without adding overhead and disrupting the Wireless Network.

We issued pings in sessions of 10 min at a ping rate of 100msec, initially, we call this aggressive scenario. The rate was defined to be 100msec to set our baseline from which we derived our sampling to obtain a suitable probing rate. The main goal is to achieve a rate which is not as aggressive as probing every 100msec.

After completing our sample analysis, we define it to be 200msec and we proceed to run test in Orbit where we can modify parameters as attenuation.

Orbit lab allow to modify attenuation from 0 dB to 30 dB. We perceived an increase in average RTT and loss rate from 27dB to 29dB. (At 30 dB link is unusable).

The results are show in the following plots.

Include Plots with Avg RTTs and Loss Rate results from Orbit

5. RESULTS

In the closing section we summarize what we have achieved, similar to what we have discussed at the closing of the previous section, 4.

Based on the tool and the methodology we used we outline the results we obtained.

What are our results telling us?

Can we identify impairments from the chosen metrics?

Which of the two methods, active or passive, can be considered to best suit the detection of Wireless impairments?

Why is the chosen method more suitable?

Future Work can be mentioned to describe the integration of this work with the project with Princeton.

6. REFERENCES

[1] Diego Da Hora, Karel Van Doorselaer, Koen Van Oost, and Renata Teixeira. *Predicting the effect of home Wi-Fi quality on QoE*. PhD thesis, INRIA; Technicolor; Telecom ParisTech, 2018.

[2] Diego Neves da Hora, Karel Van Doorselaer, Koen Van Oost, Renata Teixeira, and Christophe Diot. Passive wi-fi link capacity estimation on commodity access points. In *Traffic Monitoring and Analysis Workshop (TMA) 2016*, 2016.

[3] Diana Joumblatt, Renata Teixeira, Jaideep Chandrashekar, and Nina Taft. Hostview: Annotating end-host performance measurements with user feedback. *ACM SIGMETRICS Performance Evaluation Review*, 38(3):43–48, 2011.

[4] Partha Kanuparthy, Constantine Dovrolis, Konstantina Papagiannaki, Srinivasan Seshan, and Peter Steenkiste. Can user-level probing detect and diagnose common home-wlan pathologies. *ACM SIGCOMM Computer Communication Review*, 42(1):7–15, 2012.

[5] Kyung-Hwa Kim, Hyunwoo Nam, and Henning Schulzrinne. Wislow: A wi-fi network performance troubleshooting tool for end users. In *INFOCOM*, pages 862–870, 2014.

[6] Aaron Leonard and Shankar Ramanathan. How to get your 8821/792x wireless phones performing reliably, Jul 2018.

[7] Maximilian Ott, Ivan Seskar, Robert Siraccusa, and Manpreet Singh. Orbit testbed software architecture: Supporting experiments as a service. In *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pages 136–145. IEEE, 2005.

[8] Ashish Patro, Srinivas Govindan, and Suman Banerjee. Observing home wireless experience through wifi aps. In *Proceedings of the 19th annual international conference on Mobile computing & networking*, pages 339–350. ACM, 2013.

[9] Changhua Pei, Youjian Zhao, Guo Chen, Ruming Tang, Yuan Meng, Minghua Ma, Ken Ling, and Dan Pei. Wifi can be the weakest link of round trip network latency in the wild. In *INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2016.

[10] D Raikov. On the decomposition of poisson laws. In *Dokl. Akad. Nauk SSSR*, volume 14, pages 9–12, 1937.

[11] Srikanth Sundaresan, Nick Feamster, and Renata Teixeira. Measuring the performance of user traffic in home wireless networks. In *International Conference on Passive and Active Network Measurement*, pages 305–317. Springer, 2015.

[12] Srikanth Sundaresan, Nick Feamster, and Renata Teixeira. Home network or access link? locating last-mile downstream throughput bottlenecks. In *International Conference on Passive and Active Network Measurement*, pages 111–123. Springer, 2016.