# Performance Evaluation in 802.11ac wireless networks

*A report*
*Submitted in the partial fulfilment of requirements for the degree of*

Bachelor of Technology

in

Computer Science and Engineering


by
**Sunny Dhamnani**
Roll No. 12CS30037

Under the supervision of

**Dr. Arobinda Gupta**
Dept. of Computer Science and Engineering

and

**Dr. Sandip Chakraborty**
Dept. of Computer Science and Engineering

Indian Institute of Technology Kharagpur
Kharagpur-721302, India

# Contents

1. Abstract ........ 3
2. Introduction ........ 4
   2.1 Motivation ........ 4
   2.2 Thesis Overview ........ 4
3. 802.11 ac wireless networks ........ 5
   3.1 Better MIMO: ........ 6
   3.2 Fatter Streams: ........ 6
   3.3 5GHz Band ........ 7
   3.4 Channel Bonding ........ 7
4. Related works ........ 8
5. Details of experiment ........ 9
   5.1 System Specifications ........ 9
   5.2 Stages of Experiment ........ 10
      5.2.1 Router firmware source code compilation ........ 10
      5.2.2 Flashing the image file to the router ........ 10
      5.2.3 Setting up the traffic generator ........ 10
      5.2.4 Setting up the network ........ 11
      5.2.5 Designing experiments and recording results ........ 12
6. Observations and analysis ........ 13
   6.1 Experiment 1 - TCP throughput vs bandwidth ........ 13
   6.2 Experiment 2 - UDP throughput vs data rate ........ 14
   6.3 Experiment 3 - Jitter vs data rate ........ 15
   6.4 Experiment 4 - Packet loss vs data rate ........ 16
7. Future work plan ........ 17
8. Bibliography ........ 18

# 1. ABSTRACT

Today most of the applications involve networking, so data has to be transmitted heavily among the devices and with this new technologies for networking evolve; following Ethernet which has been a benchmark and most celebrated technology of computer networking has been the advancing wireless technology, the latest being 802.11 ac. Since 802.11 ac is relatively new area of the network so not much about the technology performance is known yet, so this is where the study of this project is focused on. This study does not aim to present any concrete reasoning rather it should be considered an empirical outcome and a step to raise questions and motivate research in this field.

In computer networking, transport layer provides end-to-end communication services to the applications. In internet there are predominantly two protocols at the transport layer

- TCP (Transmission Control Protocol) for connection-oriented transmissions
- UDP (User Datagram Protocol) for connectionless transmissions

The focus of the project is to study the performance of these protocols by changing the behavior of the network at the link layer or MAC layer. The performance is measured via various parameters namely throughput, jitter and percentage of packets lost. The immediate application of these findings can be to design efficient protocols or tweak already existing protocols to improve network performance.

# 2. Introduction

In any network structure the routers are the most active participants in data communication and with advent of technology many various routers are available commercially. Companies also claim certain data-rates which in practice are far from reality, so we try to look over how the network performs in real scenario and will try to reason why ideal data rates are not achievable in real world setting.

## 2.1 Motivation

The router used for the experimentation is Asus Router (model RT AC3200) with Merlinwrt firmware manually flashed in it which makes it more configurable. We will look at performance measures in various settings of the network at the link layer. This will enable the users of the network modify parameters to increase overall performance. Also our interest lies in finding out how protocols at higher layers like Transport Layer are to be modified to better support low layer for overall better performance, like synchronization between TCP congestion control algorithm and loss at lower level can be thought of as a potential instance of our interest.

## 2.2 Thesis Overview

In next section we will present the details of Wireless networks, 802.11 ac in specific. Also we will contrast between 802.11 n and 802.11 ac so that specific features of 802.11 ac can be appreciated. In the subsequent section we will talk about related works in the field of 802.11 ac wireless networks, then we will describe the network and system configuration that we used in our experimentation. Further we will talk briefly about the traffic generator and its configuration in our scenario, with this we will describe the details of the experiment followed by observation and analysis. Finally we will conclude with the future work plans and bibliography.

# 3. 802.11 AC WIRELESS NETWORKS

To understand meaning of the term “wireless networks” let us dissociate each term and understand the underlying concept:

- ***Wireless****:* Something in which communication occurs without the use of cables/wires (electrical component) rather occurs through radio waves.
- ***Networks****:* It is nothing but a group of various network devices (nodes) connected via communication links (wired or wireless) to share information.

Now let us understand the meaning of 802.11; IEEE 802.11 is a set of media access control (MAC) and physical layer (PHY) specifications for implementing wireless local area network (WLAN) computer communication in the 2.4, 3.6, 5, and 60 GHz frequency bands. They are created and maintained by the IEEE LAN/MAN Standards Committee (IEEE 802). The base version of the standard was released in 1997, and has had subsequent amendments. The standard and amendments provide the basis for wireless network products using the Wi-Fi brand. So in a sense Wi-Fi is the implemented technology based on the standards of 802.11. The following table gives the overview of the various technologies:-

| *Standard* | *Release Date* | *Max Speed* |
|---|---|---|
| *802.11b* | 1999 | 11 Mbps |
| *802.11a* | 1999 | 54 Mbps |
| *802.11g* | 2002 | 54 Mbps |
| *802.11n* | 2007 | 72-600 Mbps |
| *802.11ac* | 2013 | 433 Mbps – 1.3 Gbps |

So it all started with 802.11 b followed by 802.11 a then g, subsequently n and now we have 802.11 ac. When we look at Max speed across these standards we can justify the advancement in technology over time, for instance stream rate in 802.11a was order of few megabits/sec and it increased to few gigabits/sec for 802.11 ac. Wi-Fi products support earlier standards for backward compatibility. This multistandard support is typically referenced as 802.11bgn (or 802.11b/g/n), 802.11agn (802.11a/g/n), and so on. Since our focus is 802.11 ac which is immediate successor of 802.11 n so we will restrict ourselves only to the discussion of 802.11 ac and 802.11n.

The following subsections describe the 802.11 ac and compares it with 802.11n

## 3.1 Better MIMO:

Earlier 802.11a/b/g networks relied on a single antenna and a single data stream. With the introduction of the 802.11n specification, Wi-Fi could harness the power of up to three antennas and streams to dramatically improve speed, range, and reliability. 802.11ac builds on these improvements with the capability to transmit to and receive from multiple users at the same time (instead of one at a time) by using multiuser MIMO (MU-MIMO) technology. 802.11ac supports up to three antennas and streams today and will be able to support up to eight antennas in the future.

## 3.2 Fatter Streams:

The number of antennas in a Wi-Fi product determines the number of streams available. These multiple streams transmit data simultaneously over different paths that are then combined at the receiving Wi-Fi device to increase the receiver signal's capturing capabilities. Both 802.11ac and 802.11n support multiple streams. However, each stream supports only 150 Mbps in 802.11n but up to 433 Mbps in 802.11ac — three times the speed per stream. The number of antennas (and therefore streams) differentiates the performance capabilities in the 802.11n and 802.11ac standards. More antennas and streams mean not only faster speeds but also fewer dead zones, fewer dropped connections, and better coverage. As a result we can save time, view smoother videos, and move around more freely with fewer interruptions. In the vernacular of the new multispeed Wi-Fi standards, the number of send and receive antennas and stream configurations are noted as 1x1 (one antenna supporting one data stream), 2x2 (two antennas sup-porting two data streams), and 3x3. The following table compares the performance of antenna and stream configurations in the 802.11n and 802.11ac standards.

| *Configuration* | *Max Speed for 802.11n* | *Max Speed for 802.11 ac* |
|---|---|---|
| *1x1* | 150 Mbps | 433 Mbps |
| *2x2* | 300 Mbps | 867 Mbps |
| *3x3* | 450 Mbps | 1.3 Gbps |

## 3.3 5GHz Band

Wi-Fi works over two spectrum bands: 2.4 GHz and 5 GHz. Older 802.11bgn uses the 2.4 GHz frequency, which is a crowded space and subject to more interference because it's widely used by cordless phones, Bluetooth devices, and even microwave ovens. The 5 GHz frequency, which 802.11a, 802.11n, and 802.11ac use, provides five times the capacity of 2.4 GHz and is considerably less crowded, which means less interference.

## 3.4 Channel Bonding

Just as a multilane highway handles more traffic than a single-lane highway, the 5 GHz band handles a lot more traffic than the 2.4 GHz band because it has more lanes, or channels. Combining, or bonding, these channels increases their capacity. 802.11n bonded two 20 MHz–wide channels into a single double-wide 40 MHz channel. 802.11ac bonds four 20 MHz channels into a super-wide 80 MHz channel. In the future, 802.11ac products will support 160 MHz wide channel

# 4. Related works

Due to ever increasing usage of networking services there has been a lot of studies on performance of wireless networks and many of them deal with 802.11n. For instance Deek et al. [1] studied the impact of channel bonding choices as well as the effects of both co-channel and adjacent channel interference on network performance. Also there has been work on analyzing the fundamental characteristics of 802.11n links and quantifying the gains of each mechanism under diverse scenarios by Shrivastava et al. [2]. In 2010 Pelechrinis et al. [3] described whether the high transmission rates advertised by 802.11n translate to links of high quality in real-world deployments.

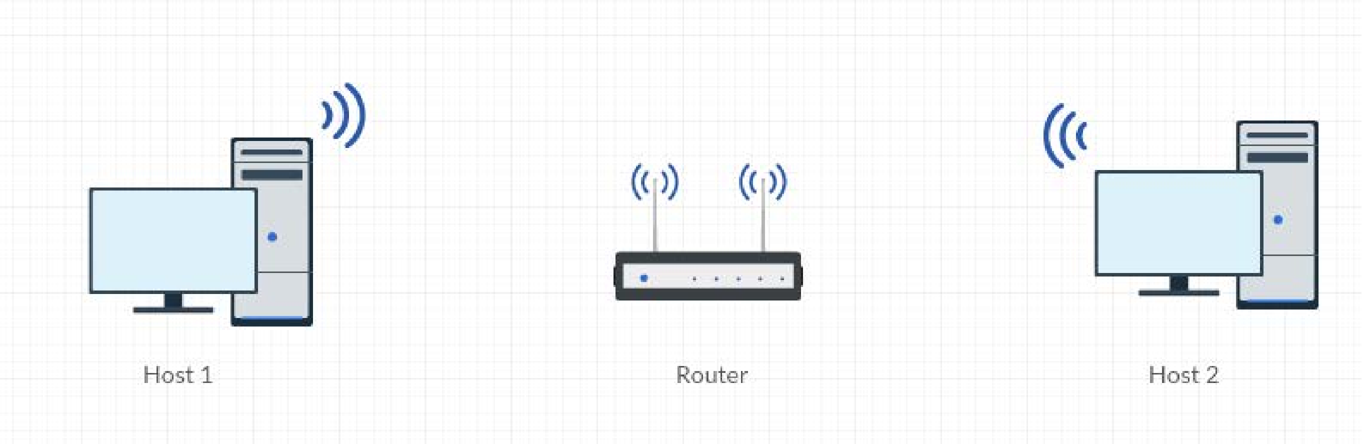
On the contrary 802.11 ac is not much researched area, in 2014 Zeng et al. [4] characterized the power consumption of 802.11ac with and without interference. There also has been investigation on the power consumption in various states of the wireless interface (sleep, idle, active) and the impact of various features of 802.11n/ac (PHY bitrate, frame aggregation, channel bonding, MIMO) on both throughput and power consumption, and the tradeoffs between these two metrics [5].

These works mentioned above are in a sense related to our current work but they also vary considerably in the experimental setting; the study by Zeng et al. [1] was mainly focused on performance taking into account the energy considerations while the work by Saha et al. [5] takes into considerations only mobile phones, which sets our work apart from current state of the art.

# 5. DETAILS OF EXPERIMENT

## 5.1 SYSTEM SPECIFICATIONS

The network setup for the experiment is depicted in the figure below.




The specifications of each of the components are mentioned below:-

| *Component* | *Details* |
|---|---|
| *HOST 1* | • OS: Ubuntu, Release 12.04(precise) 32-bit, Kernel Linux 3.13.0-66-generic<br>• Processor: Intel Core i5-4590 CPU @ 3.30GHz x 4 |
| *HOST 2* | • OS: Ubuntu, Release 14.04LTS 64-bit, Kernel Linux 3.19.0-28-generic<br>• Processor: Intel Core i5-4590 CPU @ 3.30GHz x 4 |
| *ROUTER* | • Model: RT-AC3200<br>• Firmware Version: 378.56_alpha4 powered by Asuswrt-Merlin<br>• Bootloader (CFE): 1.0.1.6<br>• Processor: ARMv7 rev 0 (v7l) (Cores 2) |

## 5.2 STAGES OF EXPERIMENT

The work done by us primarily comprised of the following stages:

1. Router firmware source code compilation
2. Flashing the compiled image file to router
3. Setting up the traffic generator
4. Setting up the network
5. Designing experiment and recording results

### 5.2.1 Router firmware source code compilation

RT-AC3200 provides a graphical interface for configuring the parameters. Through this interface channel bonding can be configured. However other parameters such as Modulation and Coding, MIMO and spatial streams are hard coded in the firmware source code. Hence to configure these, an understanding of the source code as well as its compilation is required.

An open source version (asuswrt-merlin) of the source code is available. This source code was studied in deep and sections of the code corresponding to these parameters were identified. Subsequently, we proceeded to compile the source code.

Compilation of the source code proved to be tedious and time consuming. It took nearly four weeks. The main reasons for this were:

1. Lack of proper documentation and online support
2. Very few deployments of this model
3. Version issues of the host system operating system and firmware
4. Issues with cross compilation tools

### 5.2.2 Flashing the image file to the router

The compiled image file was then dumped to router using Trivial File Transfer Protocol (TFTP) in binary mode. The crucial thing was to wait for about 10 minutes after the file has been successfully sent and then restart the router.

### 5.2.3 Setting up the traffic generator

The network performance can be analyzed by network traffic measurement in a testbed network, using a network traffic generator such as iperf, bwping etc. The traffic generator sends dummy packets, often with a unique packet identifier, making it possible to keep track of the packet delivery in the network. In our experiments we used iperf as our network traffic generator. In specific we used iperf3 which is a rewrite of iperf from scratch, with the goal of a smaller, simpler code base and a library version of the functionality that can be used in other programs. The first iperf3 release was made in January 2014. Also iperf3 is not backwards compatible with iperf2.x. In the experiment we had two hosts; one acting as a server and other as client. Client generated traffic and server acknowledged traffic and parameters were reported at the server side. Using traffic generator we changed the rate of data generation, it was typically varied from {10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 500, 800, 1000} Mbps. The command specifically used were as follows:-

- TCP
  - Command Server: iperf -s
  - Command Client: iperf -c <IP>
- UDP
  - Command server: iperf -su
  - Command Client: iperf -u -c <IP> -b <data-rate>

### 5.2.4 Setting up the network

Host 1 was the server in our experiments while Host 2 served the purpose of the client. Client generated traffic that was passed through the router. Host 1 was assigned a static IP 192.168.1.x (x=2, 6) 2 for 2.4 GHz channel and 6 for 5 GHz channel, host 2 was assigned a static IP 192.168.1.y (y=3, 7) 3 for 2.4 GHz channel and 7 for 5 GHz channel and the router has default IP 192.168.1.1. The gateway of both the hosts was set to the router’s IP i.e. 192.168.1.1 to ensure that all the traffic passed via this router. The hosts were kept more or less at equal distance from the router and were connected to router using adapter which supports 802.11ac standard. These drivers of these adapters were installed manually and were connected via USB port of the hosts.

### 5.2.5 Designing experiments and recording results

#### 5.2.5.1 *Experiment 1*

Observing impact of frequency on TCP Throughput keeping other parameters fixed. Frequency values: 2.4GHz-20MHz, 2.4GHz-40MHz, 5GHz-20MHz, 5GHz-40MHz, 5GHz-80MHz.

#### 5.2.5.2 *Experiment 2*

Observing impact of UDP Data Rate on Throughput for a fixed frequency. Data Rate is varied and its effect on throughput is observed keeping a particular frequency.

#### 5.2.5.3 *Experiment 3*

Observing impact of UDP Data Rate on Jitter for a fixed frequency. Data Rate is varied and its effect on Jitter is observed keeping a particular frequency.

#### 5.2.5.4 *Experiment 4*

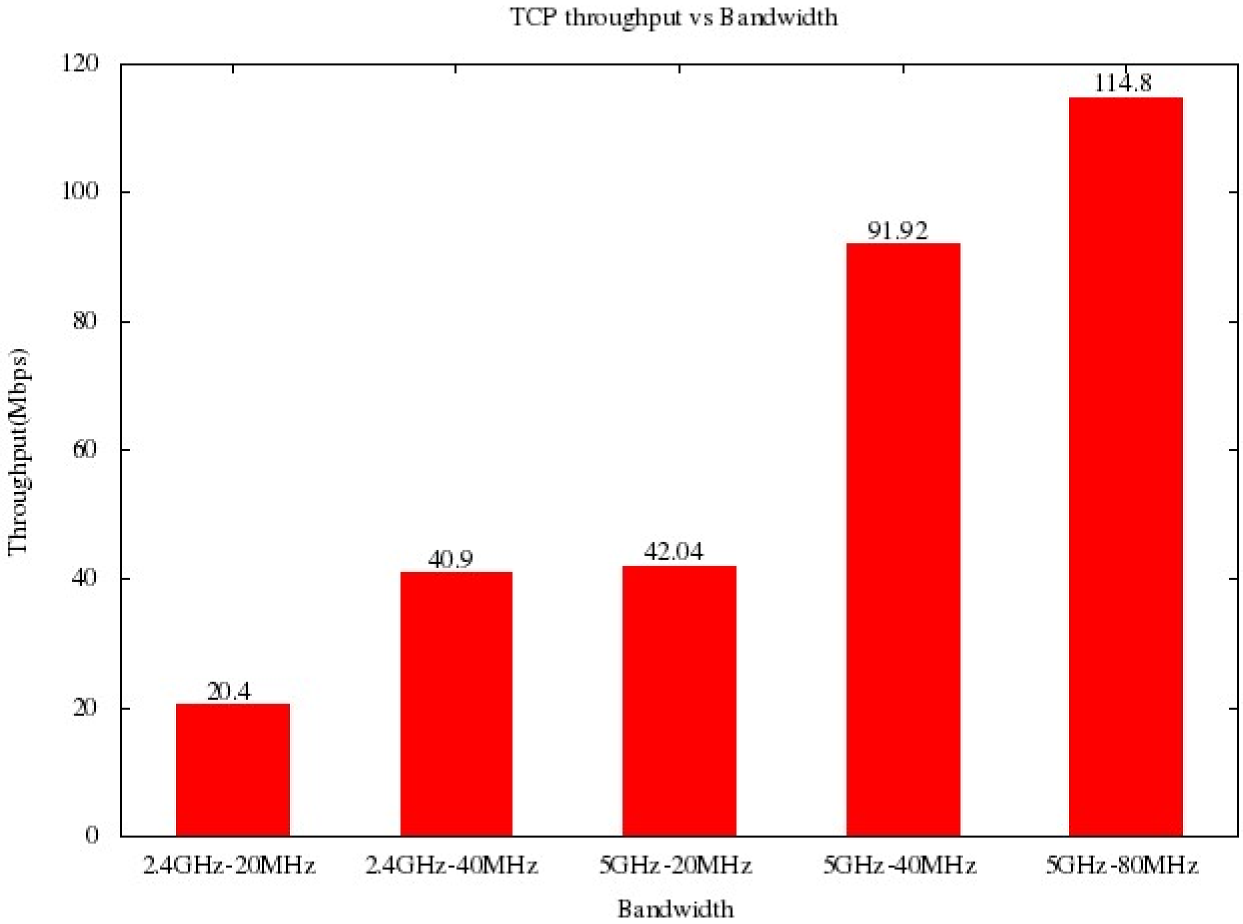
Observing impact of UDP Data Rate on packet loss for a fixed frequency. Data Rate is varied and its effect on packet loss is observed keeping a particular frequency.

The table below shows the description of data rate values taken into consideration for Experiments 2, 3 and 4

| *Frequency* | *Data Rate(Mbps)* |
|---|---|
| *2.4GHz-20MHz* | 10, 20, 30, 32, 34, 36, 38, 40, 42, 44, 47, 50, 100, 200, 500, 800, 1000 |
| *2.4GHz-40MHz* | 10, 20, 30, 40, 42, 46, 50, 52, 54, 56, 58, 60, 63, 67, 70, 100, 200, 500, 800, 1000 |
| *5GHz-20MHz* | 10, 20, 30, 40, 50, 55, 60, 63, 67, 70, 100, 200, 500, 800, 1000 |
| *5GHz-40MHz* | 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 133, 137, 140, 142, 144, 146, 149, 150, 160, 170, 200, 500, 800, 1000 |
| *5GHz-80MHz* | 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 153, 157, 160, 161, 162, 163, 164, 166, 167, 170, 180, 200, 500, 800, 1000 |

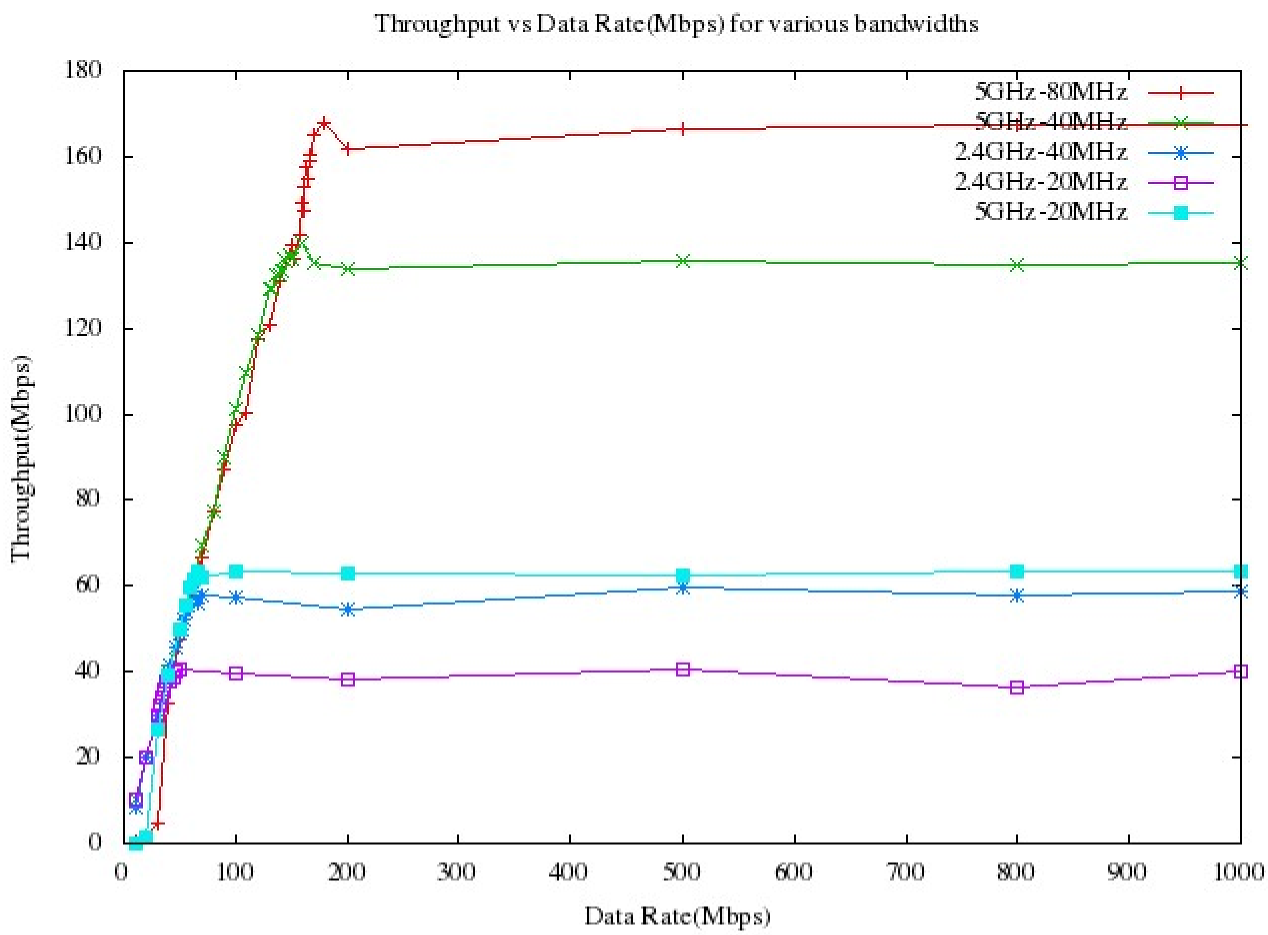
# 6. OBSERVATIONS AND ANALYSIS

## 6.1 EXPERIMENT 1 - TCP THROUGHPUT VS BANDWIDTH



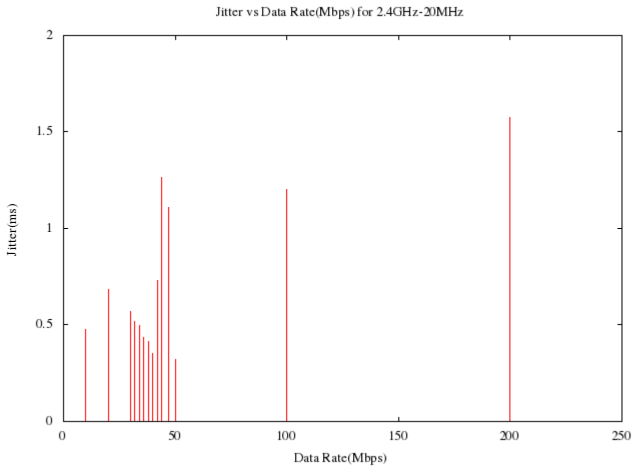

When observing the first two bars of the above graph we see that at 2.4GHz-40MHz we get double the throughput as compared to 2.4GHz-20MHz, similar relationship exist between 5GHz-20MHz and 5GHz-40 MHz. This is the expected behavior since with increase in channel width the throughput should increase in same proportion. The next observation we make here is that throughput for 2.4GHz-40MHz is close to that of 5GHz-20MHz, now this is not as expected since generally throughput is expected to be independent of central frequency. The possible reason might be because 2.4 GHz band is more crowded than 5 GHz band because it is commonly used by other wireless networks in the vicinity like Bluetooth etc. Also a wider channel increases loss impact due to wireless channel noise and because TCP's congestion control algorithm treats loss as indications of congestion, it reduces the congestion window thereby dropping the rate. Finally we can also notice that throughput for 5GHz-80MHz is not double that of 5GHz-40MHz, greater loss due to a wider channel seems to be a legit reason for this observation.

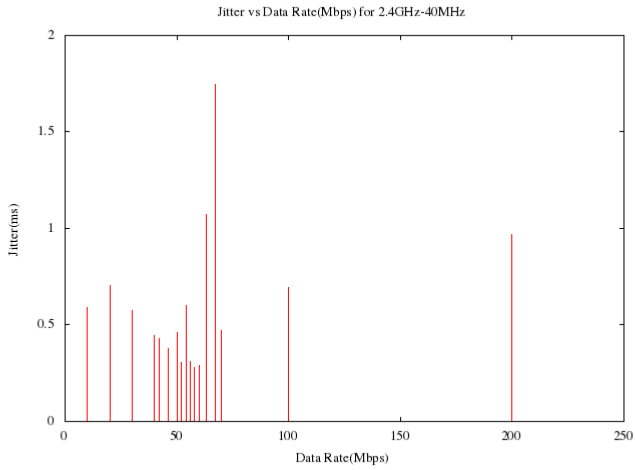
## 6.2 EXPERIMENT 2 - UDP THROUGHPUT VS DATA RATE

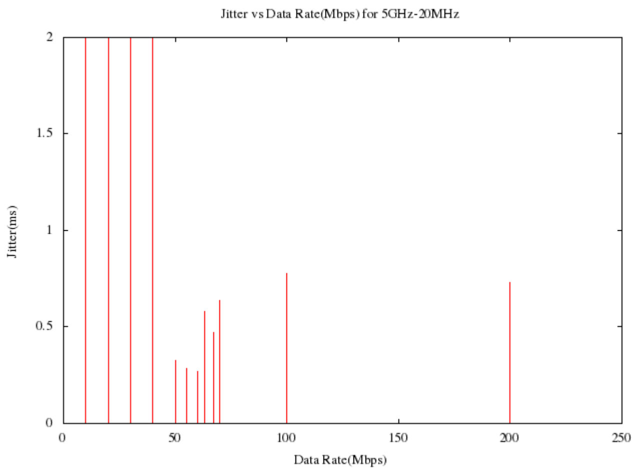


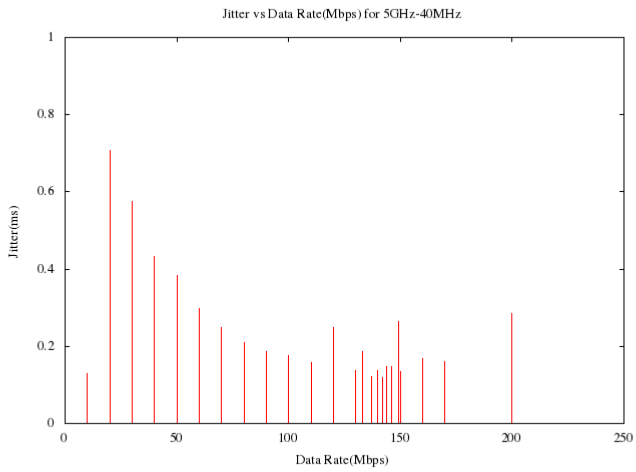

The maximum throughput achievable under considered experimental considerations is around 170 Mbps which is much less than claimed 3.2 Gbps. Goodput in UDP is greater than goodput in TCP since TCP incurs a lot of overhead due to control packets. Again here as in case of TCP 5GHz-20MHz performs similar to 2.4GHz-20MHz and a reasonable reason for this behavior is that 2.4GHz band is crowded and hence much more vulnerable to interference. Also for 2.4GHz-40MHz we get double the throughput as compared to 2.4GHz-20MHz, similar relationship exist between 5GHz-20MHz and 5GHz-40 MHz. This is the expected behavior since with increase in channel width the throughput should increase in same proportion. Finally we can also notice that throughput for 5GHz-80MHz is not double that of 5GHz-40MHz, greater loss due to a wider channel seems to be a legit reason for this observation.

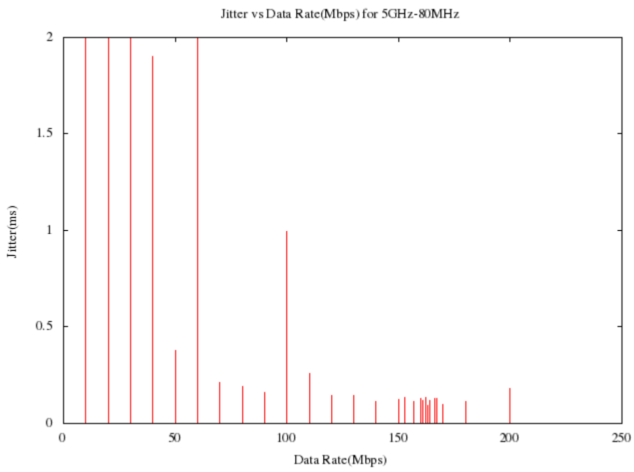
## 6.3 Experiment 3 - Jitter vs data rate



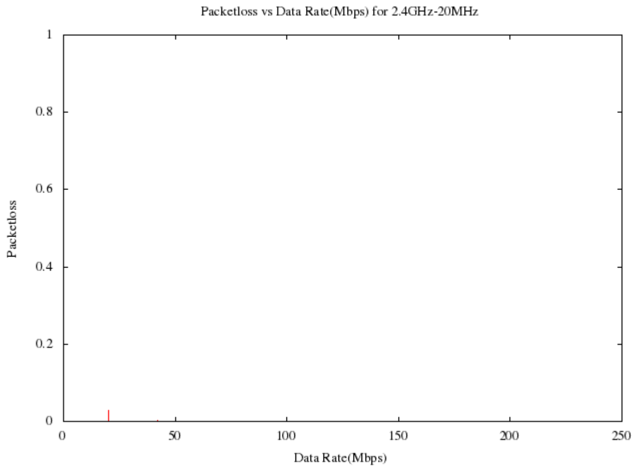



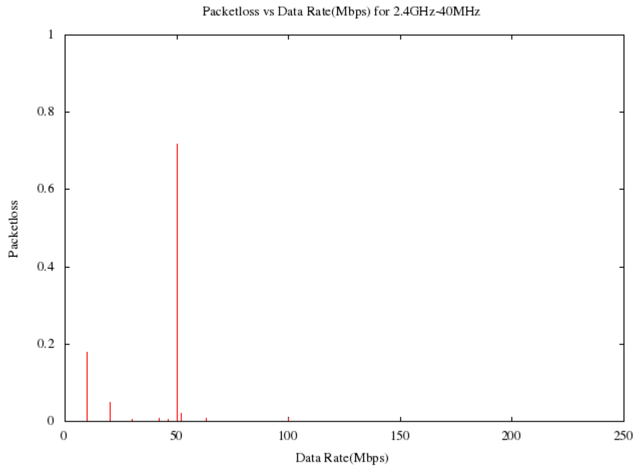



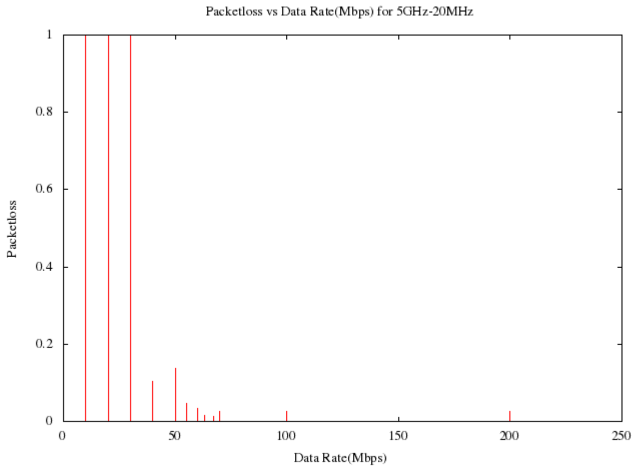



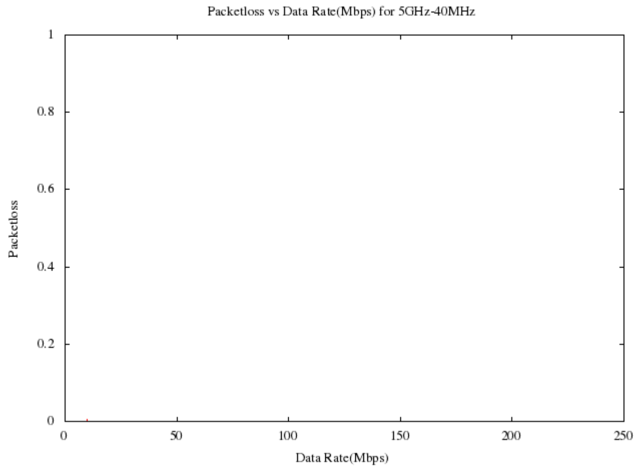

As can be seen in the above set of figures the network shows high jitter before saturation point and the reasoning behind this can be considered under future plan of action.

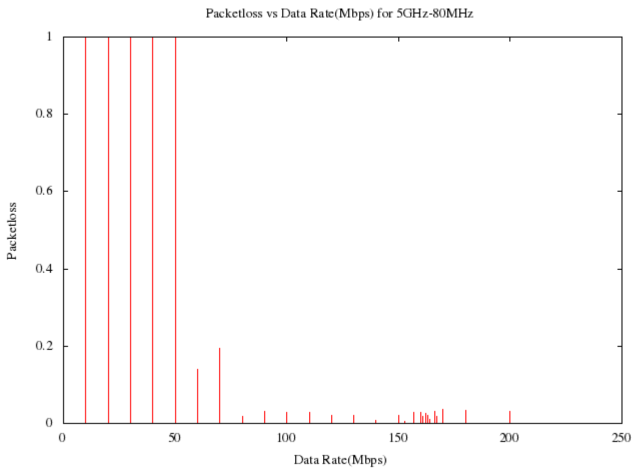
## 6.4 Experiment 4 - Packet loss vs data rate




As can be seen in the above set of figures the network shows high packet loss before saturation point and the reasoning behind this can be considered under future plan of action.

# 7. Future work plan

Although, an empirical calculation of the parameters involved is presented, the future work might include the following line of thoughts:

1. For more firm analysis we have to study more about packet transfer mechanism and may use tools like tcpdump. Also this analysis can be carried out across time of day to study the effect of interference.
2. Effect of parameters like Modulation and Coding, Spatial Streams are yet to be seen.
3. Further we can aim to modify the existing protocols to improve network efficiency in case of 802.11 ac networks.
4. Since the current work only involves two host network in future work we can make the network much more complex and study the performance with multiple nodes transmitting simultaneously

# 8. BIBLIOGRAPHY

[1] L. Deek, E. Garcia-Villegas, E. Belding, S. Lee-J, and K. Andalmeroth. *The impact of channel bonding on 802.11n network management*. Proc. ACM CoNEXT (2011).

[2] V. Shrivastava, S. Rayanchu, J. Yoon, and S. Banerjee. *802.11n under the microscope*. In IMC ’08 (2008).

[3] E. Gelal, K. Pelechrinis, I. Broustis, S. Krishnamurhty, S. Mohammed, A. Chockalingam, and S Kasera. *On the impact of MIMO diversity on higher layer performance*. In IEEEICDCS (2010).

[4] Y. Zeng, P. H. Pathak, and P. Mohapatra. *A First Look at 802.11ac inAction: Energy Efficiency and Interference Characterization.* In Proc.IFPF Networking (2014).

[5] S. K. Saha, P. Deshpande, P. P. Inamdar, R. K. Sheshadri, and D. Koutsonikolas. *Power-Throughput tradeoffs of 802.11n/ac in Smartphones.* In INFOCOM (2015).