# Implementation Of Fair Scheduler For WiFi in ns-3

*A Project Report*

*submitted by*

**HARSH RANJAN (EE16M081)**

*in partial fulfilment of the requirements*
*for the award of the degree of*

**MASTER OF TECHNOLOGY**

**in**

**COMMUNICATION AND SIGNAL PROCESSING**

**DEPARTMENT OF ELECTRICAL ENGINEERING**
**INDIAN INSTITUTE OF TECHNOLOGY MADRAS.**

**JUNE 2018**

# THESIS CERTIFICATE

This is to certify that the thesis titled **IMPLEMENTATION OF FAIR SCHEDULER FOR WIFI IN ns-3**, submitted by **Harsh Ranjan**, to the **Indian Institute of Technology, Madras**, for the award of the degree of **Master Of Technology in Communication Signal Processing**, is a bonafide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

**Dr. Venkatesh Ramaiyan**
Project Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: June 2018

# ACKNOWLEDGEMENTS

First and foremost I would like to express my indebtedness and deepest sense of gratitude to my esteemed project guide, Dr. Venkatesh Ramaiyan for introducing me to the fascinating world of WiFi and for his valuable guidance and motivation throughout the project. His constant support and advice has helped me to complete this project.

I would like to thank all professors of EE department for helping me to improve my knowledge of the related area in their lectures and discussions during course work.

I express my thanks to all my friends and classmates of IIT Madras for supporting me throughout the course. My special thanks to Kalpalatha, Ansil and Chirag who has been very supportive during the course of project.

My acknowledgement would be incomplete without thanking the biggest source of my strength, my family. I am thankful to my parents and specially my brother for their unconditional love, encouragement and support, throughout my academic carrier.

I would like to thank God Almighty for giving me the strength, knowledge, ability and opportunity to undertake this project and to persevere and complete it satisfactorily.

Harsh Ranjan

# ABSTRACT

KEYWORDS: Network Simulator ns-3, Virtual Finish Time, Weighted Fair Queue, Fair Scheduler, Minstrel



The 802.11 performance anomaly prevents modern WiFi from reaching its potential. The available Bandwidth is not fully utilized due to this performance anomaly.

To remedy this issue, we present a novel solution: we design a novel queue management scheme in network simulator ns-3. We develop a new intermediate queue with Weighted Fair Queue Scheduler that operates at the access point and doesn't require any changes to clients.

We evaluate our proposed solution for different simulation setups and examine the throughput performance of competing stations. We show that our solution achieves nearly perfect fairness for TCP and UDP downlink traffic.

# TABLE OF CONTENTS


**ACKNOWLEDGEMENTS** **i**

**ABSTRACT** **ii**

**LIST OF TABLES** **v**

**LIST OF FIGURES** **vi**

**ABBREVIATIONS** **vii**

**1 INTRODUCTION** **1**

1.1 IEEE 802.11 Performance Anomaly . . . . . . . . . . . . . . . . . . 1

1.1.1 Long-Term Equal Channel Access Probability . . . . . . . . 2

1.2 Medium Access Control(MAC) Techniques . . . . . . . . . . . . . 4

1.2.1 Carrier Sense Multiple Access With Collision Detection (CSMA/CD): . . . . . . . . . . . . . . . . . . . . . . . . . 6

1.3 Distributed Coordination Function (DCF): . . . . . . . . . . . . . . 6

**2 SCHEDULER** **11**

2.1 Packet Scheduler . . . . . . . . . . . . . . . . . . . . . . . . . . 11

2.1.1 Packet Drop Policies . . . . . . . . . . . . . . . . . . . . . 12

2.1.2 Admission Control . . . . . . . . . . . . . . . . . . . . . . 12

2.1.3 Fairness . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

2.2 Generalized Processor Sharing (GPS) . . . . . . . . . . . . . . . . 13

2.2.1 Virtual Finish Time . . . . . . . . . . . . . . . . . . . . . . 14

2.3 Weighted Fair Queue . . . . . . . . . . . . . . . . . . . . . . . . 16

**3 NETWORK SIMULATOR ns-3** **17**

3.1 WiFi Model in ns-3 . . . . . . . . . . . . . . . . . . . . . . . . . 17

3.1.1 Wifi Module in ns-3 . . . . . . . . . . . . . . . . . . . . . 17

3.2 WiFiNetDevice Architecture . . . . . . . . . . . . . . . . . . . . 18

**4 Proposed Architecture And Algorithm** **19**

4.0.1 Why Intermediate Queue at WiFiAP node? . . . . . . . . . . 19

4.0.2 storePacket(mac,packet,dest addr) . . . . . . . . . . . . . . . 20

4.0.3 callEnque() . . . . . . . . . . . . . . . . . . . . . . . . . . . 22

**5 SIMULATION RESULTS** **23**

5.1 Throughput Of Two Clients with WFQ . . . . . . . . . . . . . . . . . 23

5.1.1 UDP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

5.1.2 TCP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25

5.2 Throughput Of Clients vs Packet Threshold of MAC Queue . . . . . 26

5.3 Log Distance Path Loss Model . . . . . . . . . . . . . . . . . . . . . 27

5.3.1 Fairness Among Clients : Minstrel . . . . . . . . . . . . . . . 28

**6 CONCLUSION AND FUTURE WORKS** **31**

6.1 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31

6.2 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31

**7 REFERENCES** **32**

# LIST OF TABLES


5.1 simulation setup of WFQ for UDP . . . . . . . . . . . . . . . . . . . 23

5.2 UDP Downlink Throughput of Two Clients with different weights . 24

5.3 simulation setup of WFQ for TCP . . . . . . . . . . . . . . . . . . . 25

5.4 TCP Downlink Throughput of Two Clients with different weights . 25

5.5 simulation setup3 . . . . . . . . . . . . . . . . . . . . . . . . . . . 26

5.6 RSSI-MCS Mapping . . . . . . . . . . . . . . . . . . . . . . . . . . 27

5.7 Distance-RSSI-MCS-Datarate Mapping . . . . . . . . . . . . . . . . 28

5.8 Simuation setup for 1 server 1 AP 5 Clients for both case1 and case2 28

5.9 Clients Details for Case1 . . . . . . . . . . . . . . . . . . . . . . . 29

5.10 Clients Details for Case2 . . . . . . . . . . . . . . . . . . . . . . . 30

# LIST OF FIGURES


| | | |
|---|---|---|
| 1.1 | An AP with two clients in a cell | 2 |
| 1.2 | Two clients with different data rate | 3 |
| 1.3 | OSI Reference Model and IEEE 802.11 | 4 |
| 1.4 | Possible MAC Techniques | 5 |
| 1.5 | `Four-Way Handshake Protocol` | 7 |
| 1.6 | IEEE 802.11 DCF basic access method | 8 |
| 1.7 | Possible MAC Techniques | 9 |
| 1.8 | IEEE 802.11 DCF RTS/CTS method | 9 |
| 2.1 | A Typical Wireless Scheduler | 11 |
| 2.2 | Max-min Fairness | 13 |
| 2.3 | Generalized Processor Sharing | 14 |
| 2.4 | Virtual Finish time V(t) and GPS and WFQ Scheduler Output | 15 |
| 2.5 | Weighted Fair Queue | 16 |
| 3.1 | Architecture of WifiNetDevice | 18 |
| 4.1 | WiFiIntQueue at AP | 19 |
| 5.1 | 1 Server 1 AP and 2 Clients for WFQ Experiment | 23 |
| 5.2 | UDP Downlink throughput of two clients | 24 |
| 5.3 | TCP Downlink throughput of two clients | 25 |
| 5.4 | UDP Downlink throughput vs packet threshold of mac-queue | 26 |
| 5.5 | UDP Downlink throughput of 5 clients for Case 1 | 29 |
| 5.6 | UDP Downlink throughput of 5 clients for Case 2 | 30 |

# ABBREVIATIONS

| | |
|---|---|
| **WLAN** | Wireless Local Area Network |
| **DCF** | Distributed Coordination Function |
| **CSMA** | Carrier Sense Multiple Access |
| **CA** | Collision Avoidance |
| **SIFS** | Short Inter Frame Space |
| **PIFS** | Point Coordination Inter Frame Space |
| **DIFS** | Distributed Inter Frame Space |
| **EIFS** | Extended Inter Frame Space |
| **GPS** | Generalized Processor Sharing |
| **VFT** | Virtual Finish Time |
| **WFQ** | Weighted Fair Queue |
| **FCFS** | First Come First Serve |
| **ATF** | AirTime Fairness |
| **TCP** | Transmission Control Protocol |
| **UDP** | User Datagram Protocol |

# CHAPTER 1

# INTRODUCTION

Wireless Fidelity (WiFi) is a technology for wireless local area networking with devices based on IEEE 802.11 standards. It is designed to provide in-building broadband coverage. Current WiFi systems support a peak physical-layer data rate of 54 Mbps and typically provide indoor coverage over a distance of 100 feet. WiFi has become the de facto standard for last mile broadband connectivity in homes, offices, airports, railway stations, parks, restaurants, hospitals and public hotspot locations. It can usually provide a coverage range of only about 100 meters from the access point. IEEE 802.11 is an emerging IEEE standard for broadband Wireless LANs (WLANs) . Fairly sharing the wireless communication medium or resources among all connected stations is the main issue. The protocols used for sharing the medium in fair way are called coordination functions. There are three types of coordination function: (i) Basic DCF (Distributed Coordiantion Function), (ii) DCF with RTS/CTS (Request to Send/Clear to Send) and (iii) PCF (Point Coordination Function)

First two, basic DCF and DCF with RTS/CTS, coordination function are modeled in the context of ad-hoc mode and latter is in the context of infrastructure mode where access point (AP) is present.

The objective of this project is to design a novel queue management scheme to remedy the issue of performance anomaly in WiFi network.

## 1.1 IEEE 802.11 Performance Anomaly

In a multirate wireless network, clients or stations which are connected to the same AP (SSID) can use different data rates. They have their choice to transmit the data at varying data rates. For example, in IEEE 802.11b protocol, client can transfer at 1Mbps, 2Mbps, 5.5Mbps, or 11Mbps data rate.

A station uses an "**auto-rate**" mechanism to select an appropriate data rate for transmission based on the channel condition experienced. In a wireless LAN network, some hosts may be far away from their Access Point(AP). When the 802.11 MAC detects a packet loss due to the absence of a synchronous ACK, it continues re-transmitting the packet until the maximum retry limit has been reached.In such cases, the sender can transmit at a lower data rate (using a more resilient modulation scheme) so that the channel bit error rate (BER) is reduced. In general, there is a trade-off between data rate and BER in wireless networks.

so that the quality of their radio transmission is low. .This phenomenon is also known as **rate anomaly** problem in IEEE Distributed Coordination Function(DCF) networks. The reason of this cause is the fairness philosophy of 802.11 standards, which guarantees **"long-term equal channel access probability"**.

### 1.1.1 Long-Term Equal Channel Access Probability

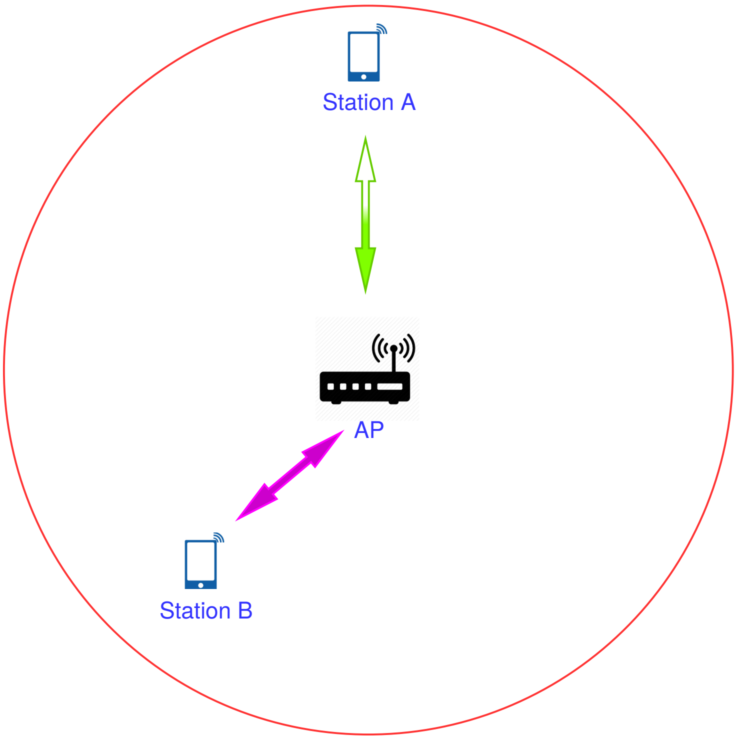



Figure 1.1: An AP with two clients in a cell

It states that if same-sized packets are transmitted under similar channel conditions, each station gets roughly equal throughput, regardless of its own transmission rate. Now, if all stations which are connected to same AP are using same data rate, IEEE 802.11 automatically ensures equal time shares among all competing stations. In this situation there is fairness to share the resources. But in the real world, this situa-

tion is very ideal. In reality competing stations can use different data rate due to either channel condition or different protocol of stations used.
For example, two stations are connected to same access point in a wireless network. And both are transmitting the data at different data rates.

Suppose station A is transmitting or receiving the data at 1 Mbps and Station B at 10 Mbps. For transmitting the similar data and in similar channel condition, station A occupies the channel for 10 times than station B does.

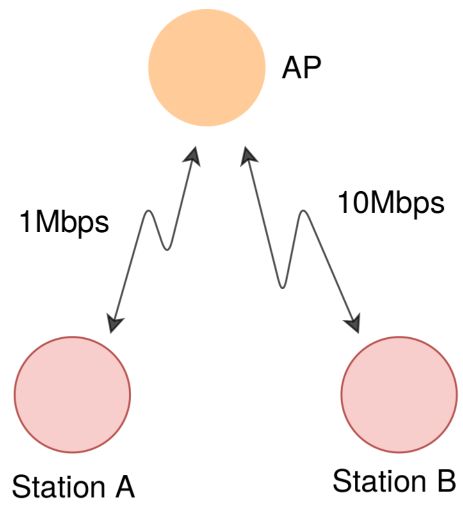



Figure 1.2: Two clients with different data rate

So, the channel occupancy ratio for two competing stations are roughly proportional to their individual data rates.

Therefore, a multi-rate wireless network privileges slower stations and penalizes faster stations, since they need to wait for the slower stations to complete their transmissions or re-transmissions.

Here, the concept of Unfairness comes to light. Now we easily see that the slower stations occupy more channel airtime to transmit the similar number of packets as faster stations.

And the experiments show that in the presence of slower station, each station achieves same throughput, irrespective of its data rate. It means that each station starts operating on the data rate that slower station has because of baseline property. Therefore the aggregate throughput of the system decreases.

**Baseline Property:** The long-term throughput of a node competing against any number of nodes running at different speeds is equal to the throughput that the node would achieve in an existing single-rate 802.11 WLAN in which all competing nodes were running at its rate.
It means, the throughput a node achieves when competing against n nodes is identical to what it would achieve if it were competing against n nodes were running at its rate.

## 1.2 Medium Access Control(MAC) Techniques

Media access control (MAC) is a sublayer of the Data Link Layer (DLL) in the seven-layer OSI network reference model. This is also known as MAC layer. The fundamental role of MAC is to provide an addressing mechanism and resource access so that each station in a network can communicate with other stations.

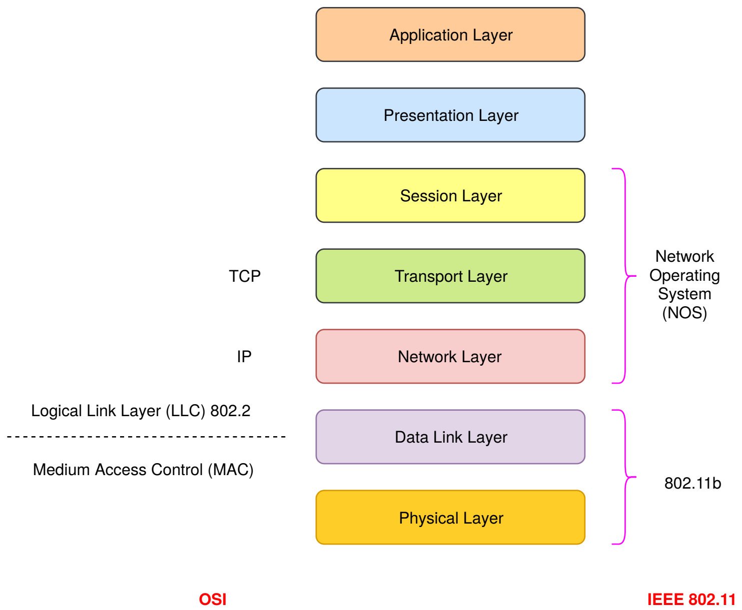



Figure 1.3: OSI Reference Model and IEEE 802.11

If there is only one station presents in the network, it can send or communicate anytime. i.e. there is no other station in the network to be competed. But when more than one station are present, there are conflicts among all stations to get the access of the resources. This problem is known as resource contention problem in wireless network. The protocol uses to resolve this problem is known as MAC techniques.

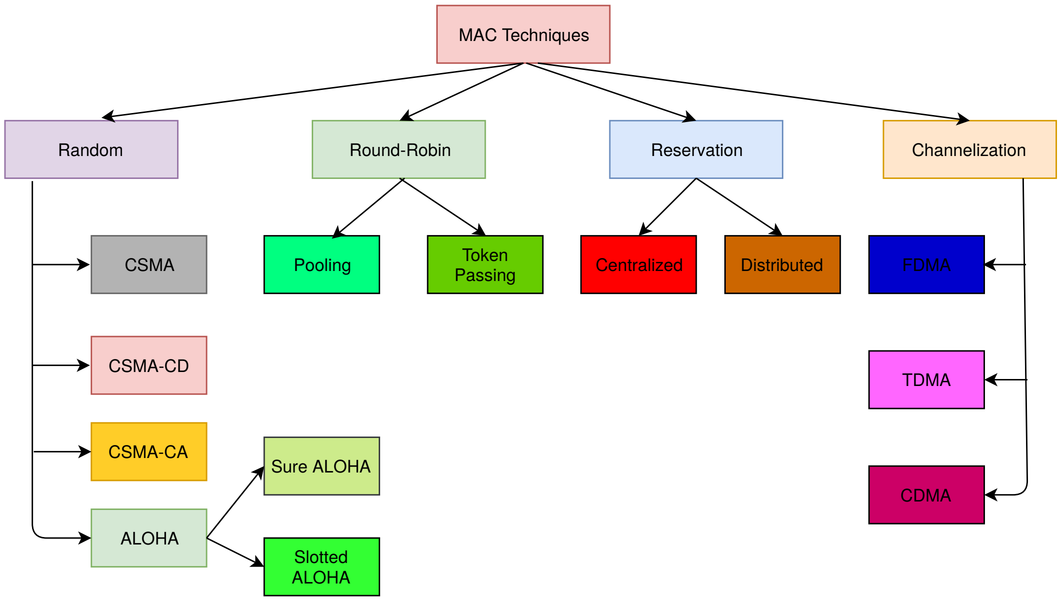



Figure 1.4: Possible MAC Techniques

The MAC decides WHERE and HOW to resolve it. WHERE refers to the control is performed in a centralized or distributed manner. In a centralized system a master station grants access of the medium to other stations. But in the distributed point of view, all stations perform a medium access control function and dynamically decide which station to be granted access. HOW refers how control is performed. There are various ways for medium access control. There are four broad categories in which MAC techniques are divided.All MAC techniques are shown in figure 1.7.

It is the responsibility of MAC to transmit the data packets to and from the network-interface card, and to and from another remotely shared channel.As we know that in multi-class network, MAC decides which station/node gets access of the resource. The main goal of distribution is MAC must distribute it fairly to each node.

### 1.2.1 Carrier Sense Multiple Access With Collision Detection (CSMA/CD):

Most of the WLAN products use Carrier Sense Multiple Access with Collision Detection (CSMA/CD) as the MAC protocol. Carrier Sense means that the station will listen or sense the channel or medium before transmitting the data. If there is already someone transmitting, then the station waits and tries again later. If no one is transferring the data, the station move forward and transmits its data. But when more than one station tries to access the channel, the transmissions will collide and the information will be lost. To avoid this problem collision detection is used. The station will sense the channel to ensure that its transmission made it to the destination without any collision. If a collision happens, the stations will wait and try again later. The waiting time of station for this is determined by the back-off algorithm. This technique works great for WLANs but wireless topologies can generate a problem for CSMA/CD. However, the wireless medium introduces some unique challenges and those are not present in WLANs. **Hidden station** and **Exposed station** problems are two important challenges from them.

These two problems are resolved out with the MAC used for IEEE 802.11 standards by using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). CSMA/CA will be discussed in more detail in next section.

## 1.3 Distributed Coordination Function (DCF):

In IEEE 802.11, the DCF is the fundamental access method used to support asynchronous data transfer . All the stations in a basic service set(BSS) must support DCF. The DCF supports Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). CSMA/CD is not used because a station is unable to listen to the channel while transmitting. In 802.11, Carrier sensing is performed at both physical layer and mac layer. It performs physical carrier sensing at PHY layer and virtual carrier sensing at MAC layer.

There are two techniques used for data transmitting in DCF:

(i) Basic Access Method and

(ii) RTS-CTS technique.

**Basic Access Method** is also known as two-way handshaking mechanism. After a successful data transmission, the destination station sends an MAC acknowledgement (ACK) to source station.

**RTS-CTS Method** is a four-way handshaking mechanism, which uses Request-To-Send and Clear-To-Send technique to retain the channel before data transmission. This technique is used to mitigate the performance degradation due to hidden network. But this technique increases overhead for short data frames.

The IEEE 802.11b standard supports for both distributed access mechanism and contention-free arbitration access method for media access control (MAC). DCF uses CSMA/CA for chanel access and PCF provides for contention-free access via an arbitrator that resides in access points. IEEE 802.11b standard allows both the methods to coexist: a contention period followed by a contention-free period.

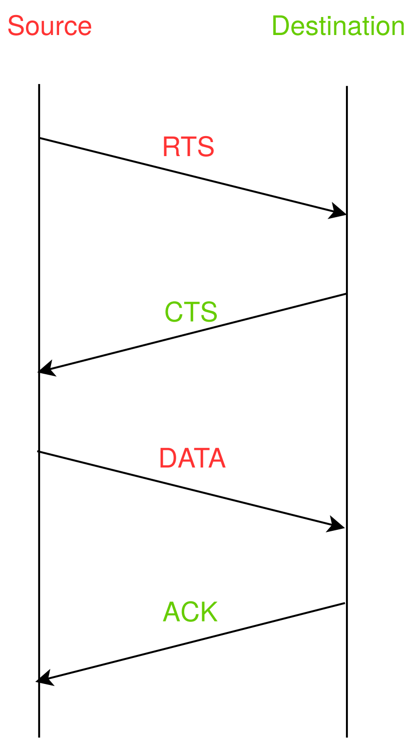



Figure 1.5: Four-Way Handshake Protocol

The DCF mechanism follows the CSMA/CA principle. A station willing to transmit the data, first scans the channel before transmitting. If the channel is idle for Distributed Interframe Space (DIFS) period of time, the station starts to transmit. Else, it continues to scan the channel until it is observed idle for DIFS period of time. It then generates a random backoff interval to minimize the probability of collision with other contending stations. A random backoff interval is decided by exponential backoff scheme. This is called collision-avoidance stage.

For every packet transmission, the backoff period is uniformly distributed between (0 and W-1). Here W is called contention window and its value depends on number of failed attempt for transmission of the packet. For the very first packet, W is initialized to minimum contention window $CW_{min}$.

$$CW_{max} = 2^m CW_{min} \tag{1.1}$$

The value of m is doubled till a maximum value of $CW_{max}$ from equation 1.1, with every unsuccessful data transmission.
The back-off counter is decremented as long as the channel is scanned idle during the collision-avoidance stage. It goes to FROZEN stage if a transmission is sensed on the channel. And it is reactivated again when the channel is scanned idle. Once the back-off counter reaches 0, the station starts its data transmission.

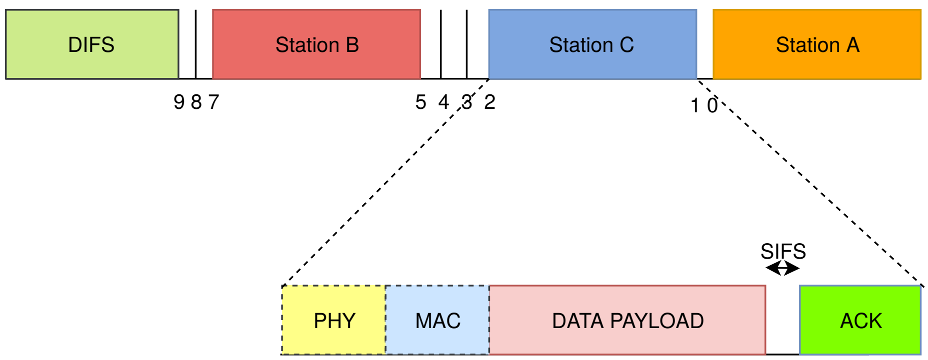



Figure 1.6: IEEE 802.11 DCF basic access method

The transmitter marks the packet to a successful reception only when it receives a positive acknowledgement from the receiver within a fixed time interval. This time interval is also known as Short Interframe Space (SIFS). If transmitter does not receive any ACK in SIFS time interval, the transmitter assumes that the collision is happened at the receiver end.

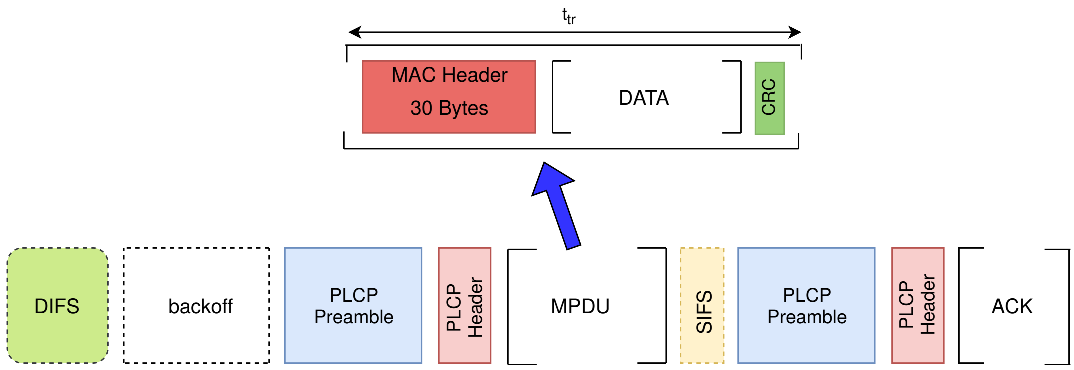



Figure 1.7: Possible MAC Techniques

In figure 1.6, the contention process for three stations is shown. All three stations A, B and C are attempting to transmit their data using the DCF procedure. Station A selects a backoff window of 9 slots. When the backoff counter goes to slot 7, it has to freeze, since station B has completed its backoff at this slot. Station A again starts its backoff once Station B has completed its transmission. Once again station A goes to freeze stage at slot 2, since Station C has completed its backoff this time. Once the station C completes its data transmission and backoff of station A again starts and this time it reaches 0. Therfore, station A gets the access of the channel and then it transmits its data. This is a example of two-way handshaking mechanism and also known as Basic Access Technique.

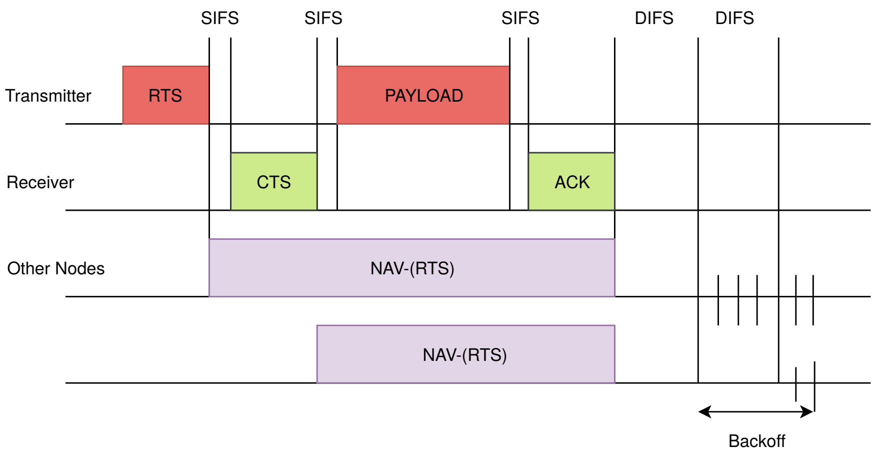



Figure 1.8: IEEE 802.11 DCF RTS/CTS method

DCF also defines an additional Four-Way handshaking procedure. It is known as

RTS/CTS mechanism. RTS stands for Request-To-Send and CTS for Clear-To-Send. This access mechanism strives to reserve the channel for the entire packet transmission duration and, hence, minimizes the impact of collisions. The additional cost for this mechanism is a slightly increased transmission overhead (i.e. RTS/CTS frame exchange). Figure 1.8 shows the RTS/CTS mechanism. Notice that the channel in the transmitterâs vicinity is reserved via an RTS and CTS packet exchange.

# CHAPTER 2

# SCHEDULER

## 2.1 Packet Scheduler

In a multi-class network, Scheduler determines the sequence in which the data should be sent to the output link so that it achieves QoS. Quality of Service(QoS) represents the set of necessary techniques to manage network bandwidth, delay, jitter and packet loss. It resolves the recourse contention problem. It fairly shares resources and provides performance guarantee. firstly it decides the order of requests to be served or not using admission control algorithm and then it manages the queues of these awaiting requests. It treats each flows differently and gives different users different QoS. The basic task of scheduling disciplines is they can allocate channel bandwidth, delay and loss. On the basis of these allocation, fairness of the network is decided.

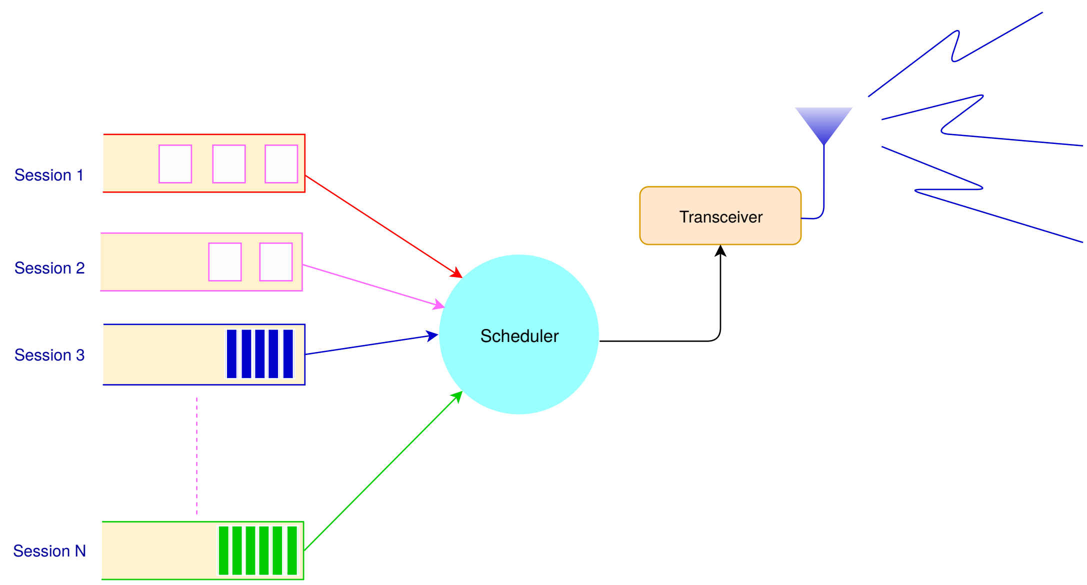



Figure 2.1: A Typical Wireless Scheduler

To achieve QoS, it is required to identify traffic sources and types. There is a need for appropriate handling of real time and non-real time traffic such as,

- Voice (Delay sensitive)

- Video (Bandwidth intensive)
- Data (Loss sensitive)
- HTTP, FTP, SMTP
- Bursty and Constant type
- Multi-service traffic: IP, MPLS
- Single or Multiple flows of the same type

The parameters which influence the QoS are Latency, Jitters and Packet Loss. Scheduling algorithms must be simple so that it can be implemented easily.

### 2.1.1 Packet Drop Policies

When a packet arrives,the scheduler invokes packet dropping algorithm to determine if the arrived packet is placed in queue or discarded.

- **Drop from tail:** The arrived packet is dropped if queue is full when it arrives.
- **Drop from head:** The head packet i.e. the oldest packet of queue is dropped if queue is full when it arrives
- **Random drop:** When a receiving buffer is congested, the scheduler randomly selects one of the packet within a queue to be dropped.
- **Replicated Copies:** When a packet arrives and buffer overflow occurs, the number of times each packet has been replicated can be used to decide which packet should be dropped. Two possible ways to drop packet: (i) the packet that has been less replicated is dropped first and (ii) the packet that has been more replicated is dropped first.
- **Remaining Lifetime:** This policy selects the packet to drop based on their remaining TTL. Two possible ways to drop packet: (i) the packet with the smallest remaining TTL is discarded first and (ii) the packet with the longest remaining TTL is discarded first.

### 2.1.2 Admission Control

Admission control decides whether a new flow can be allowed to join so that the performance of the system cannot be degraded. So it needs to provide QoS. Scheduling discipline affects the ease of admission control algorithm.

### 2.1.3 Fairness

Scheduling discipline allocates a resource to the flow. An allocation is fair if it satisfies max-min fairness algorithm. In **max-min fairness** , a fair share allocates a user with a "small" demand what it wants, and evenly distributes unused resources to the "big" users intuitively.

- Resources are allocated in order of increasing demand.
- No source gets a resource share larger than its demand.
- Sources with unsatisfied demands get an equal share of the resource.

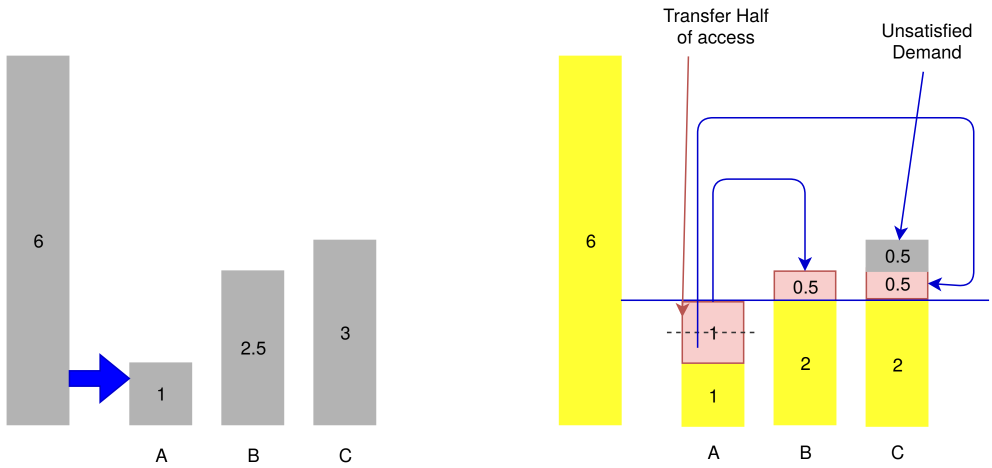



Figure 2.2: Max-min Fairness

Fairness provides protection to flows. it manages resources in such a way that traffic hogs cannot overrun others. It builds firewall around flows. It means each flow behaves independently. It also provides performance guarantee to each flow.

## 2.2 Generalized Processor Sharing (GPS)

Generalized Processor Sharing (GPS) is a service policy for multiclass system and it is a fluid fair queuing. It means

- traffic is fluid in nature.
- It serves infinitesimal amount of data from each backlogged queue.
- provides Max-min fairness.

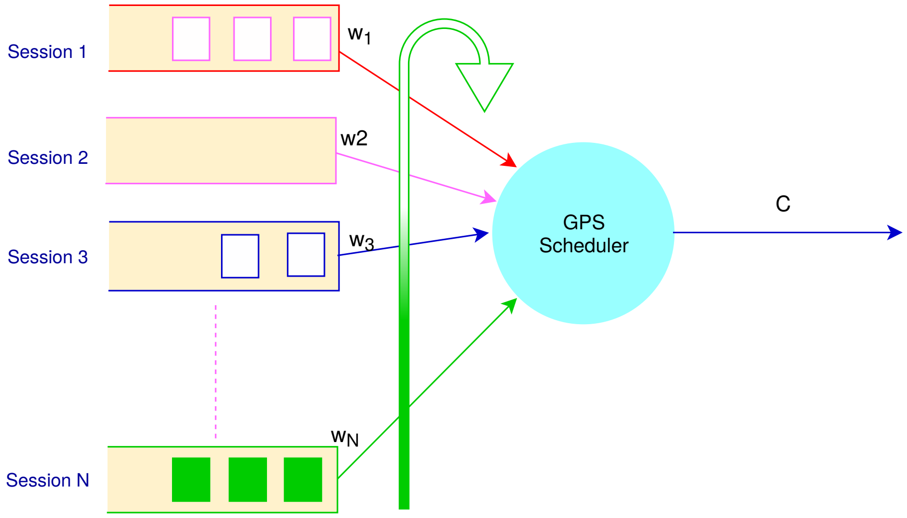



Figure 2.3: Generalized Processor Sharing

It provides both fairness and performance guarantee. Generalized Processor Sharing (GPS) is a fluid fair queuing. It means traffic is fluid in nature. It serves infinitesimal amount of data from each backlogged queue. It provides both fairness and performance guarantee. In this scheduling techniques, a weight ($\phi_i > 0$) is assigned to each class (i.e. its queue). One can interpret: the scheduler serves ($\phi_i$) bits from queue i for each ($\phi_j$) bits served from queue j, when both queue i and j are backlogged queue. The scheduler visits the queues in a **round-robin** manner and serves ($\phi_i$ ) bits from queue i in each round.

GPS is a fluid version of the bit-wise weighted round robin (WRR) scheduling. In this scheduling techniques, each backlogged queue is served simultaneously. So, it is the best scheduling technique on the basis of providing fairness. And it also gives performance to each queue. But it is impossible to implement this scheduler in reality, since it is not possible to serve infinitesimals.

### 2.2.1 Virtual Finish Time

Properties of virtual finish time V(t) :

- V(0) = 0, and V(t) $\geq$ 0 for t $\geq$ 0.
- V(t) is a non-decreasing function of time.
- V(t) is a peace-wise linear function and its slope increases whenever a queue becomes empty and decreases when an empty queue receives new data.

- The slope of V(t) is 0 whenever the system has no data to serve.

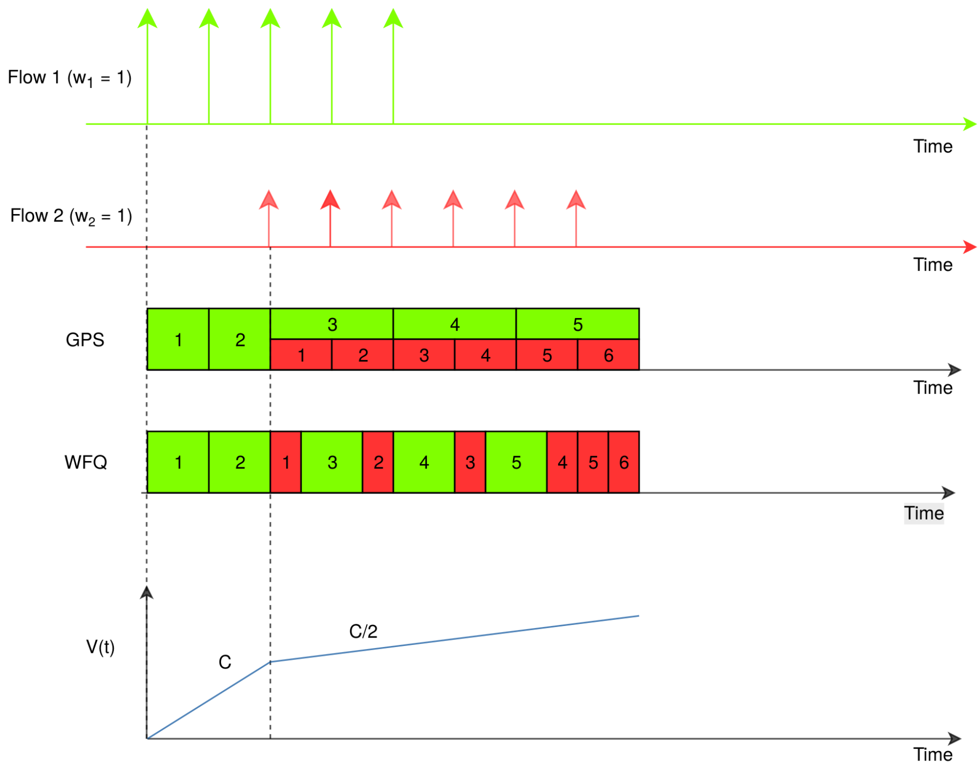



Figure 2.4: Virtual Finish time V(t) and GPS and WFQ Scheduler Output

Virtual finish time of $k+1^{th}$ packet of class j

$$\boldsymbol{V(d^j_{k+1}) = \frac{L^j_{k+1}}{\phi_j} + max(V(d^j_k), V(a^j_{k+1}))} \tag{2.1}$$

where,

$V(d^j_{k+1}) \triangleq \ virtual\ \ finish\ \ time\ \ of\ \ k+1^{th}\ \ packet\ \ of\ \ class\ \ j$

$L^j_{k+1} \triangleq \ packet\ \ length\ \ of\ \ k+1^{th}\ \ packet\ \ of\ \ class\ \ j$

$\phi_j \triangleq \ weight\ \ of\ \ class\ \ j$

$V(d^j_k) \triangleq \ virtual\ \ finish\ \ time\ \ of\ \ k^{th}\ \ packet\ \ of\ \ class\ \ j$

$V(a^j_{k+1}) \triangleq \ arrival\ \ time\ \ of\ \ k+1^{th}\ \ packet\ \ of\ \ class\ \ j$

## 2.3 Weighted Fair Queue

WFQ is the packetized version of GPS and also known as packet GPS (PGPS). It sends one packet at a time not like GPS does. It is a work-conserving scheduler. It also assigns weight to each class to divide the bandwidth across all classes and provides Max-Min fairness.

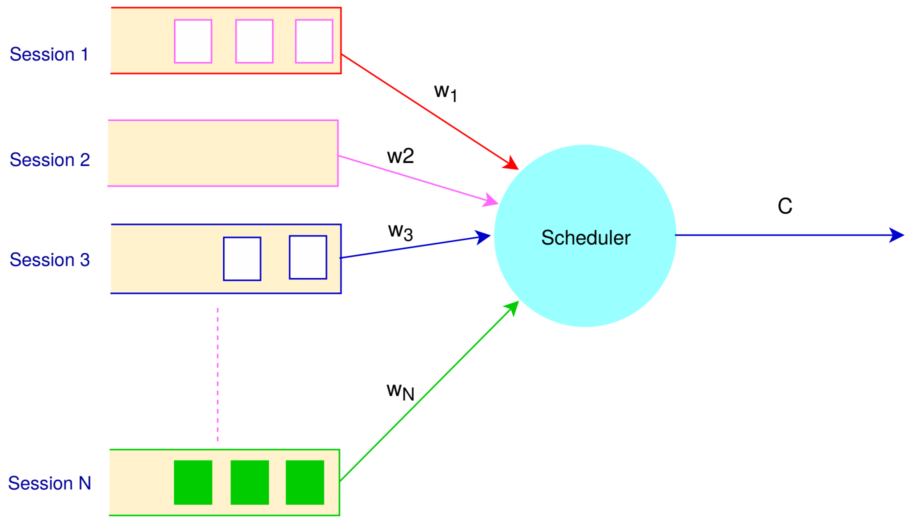



Figure 2.5: Weighted Fair Queue

WFQ scheduling technique follows following steps:

- The scheduler calculates virtual finish time of packet, as soon as packet arrives into the scheduler.
- Arrange all packets in the queue in ascending order of their virtual finish time. (Ties can be broken by giving priority to smallest index of class.)
- Scheduler sends Head Of Line (HOL) packet and it is transmitted completely.

# CHAPTER 3

# NETWORK SIMULATOR ns-3

## 3.1 WiFi Model in ns-3

ns-3 is a discrete-event network simulator. A network simulator is a technique by which the performance of the network system or network application is evaluated before using in the real world. It can analyzes the each component or devices of the network. ns-3 is used in research and teaching. Its libraries are written in c++ with Python bindings. It is a free software, licensed under the GNU GPLv2 license, developed with a focus on research and education. The latest version of the simulation software, ns-3.28, provides support for a variety of protocols and standards in the data link layer (e.g., models for WiFi, LTE), network layer (supports both IP and non-IP based networks, routing protocols such as OLSR, AODV) and transport layer (variants of TCP, UDP). In addition, the software includes modules for application layer protocols, mobility, help for configuring a variety of network systems as well as tracing modules for measurements and analytics. The simulation software also has a real time scheduler for interaction with real systems and supports reuse of real application and kernel code. Several external animators and data analysis and visualization tools like NetAnim and Wireshark are used with ns-3 for better understanding and analysis.

### 3.1.1 Wifi Module in ns-3

ns-3 provides WiFiNetDevice model for a wireless network NIC(Network Interface Card) based on IEEE 802.11 standards.
This model provides following aspects of 802.11

- Basic 802.11 DCF with infrastructure and ad-hoc modes,
- QoS-based EDCA and queueing extensions of 802.11e,
- MSDU and MPDU aggregation extensions of 802.11n,

- Supports 802.11a, 802.11b, 802.11g, 802.11n (both 2.4 and 5 GHz bands) and 802.11ac
- Various rate control algorithms including Aarf, Arf, ConstantRate and Minstrel etc.

Nodes can have many WiFiNetNetevice based on the channel required.

## 3.2 WiFiNetDevice Architecture

WiFi model is divided into three sublayers

- PHY Layer Model,
- MAC High Model and
- MAC Low Model

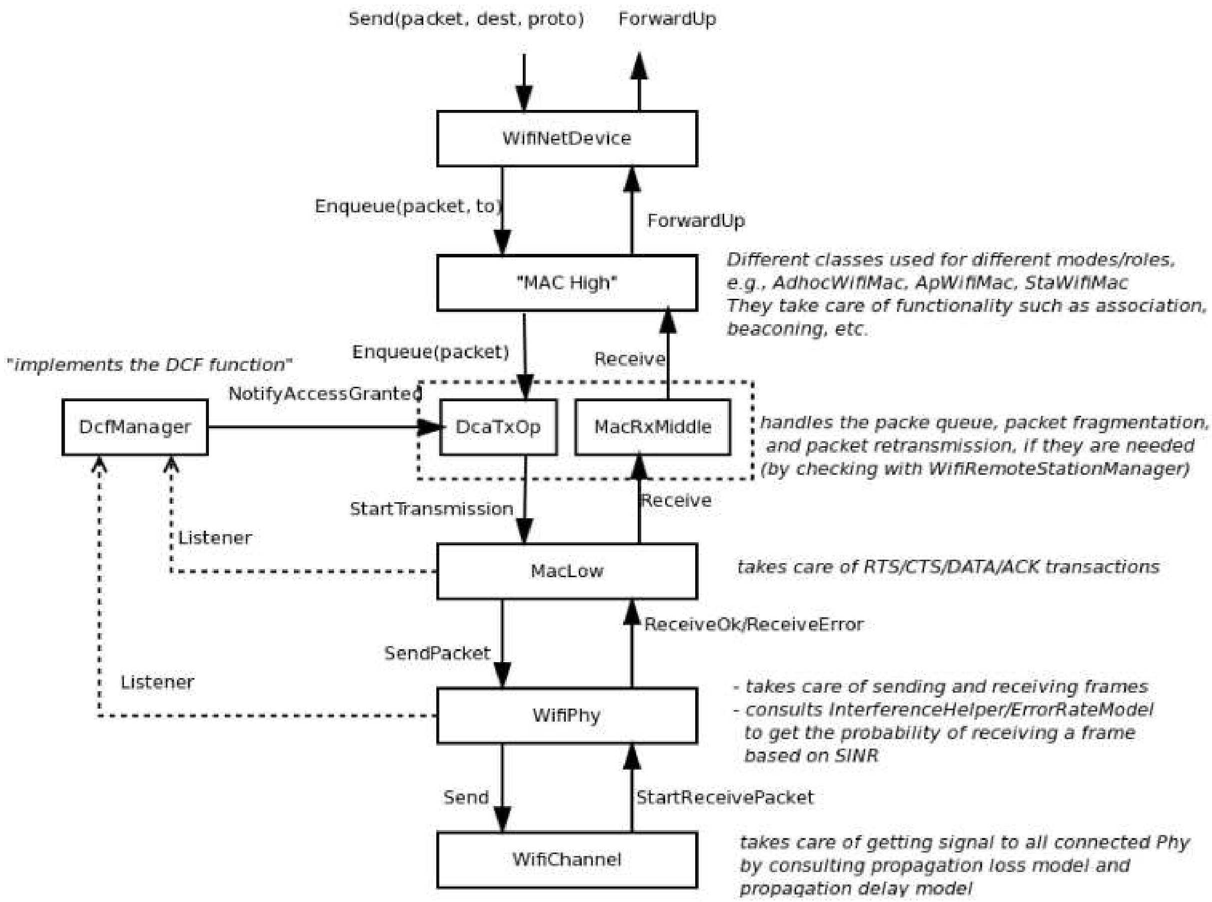



Figure 3.1: Architecture of WifiNetDevice

# CHAPTER 4

# Proposed Architecture And Algorithm

WiFiNetDevice directly sends the packet to MAC queue with mac and its destination address as parameters. It is shown in figure 4.1. Destination address is the mac address of a client. Access Point(AP) sends beacon with destination address of ff:ff:ff:ff:ff:ff. WifiIntQueue is installed on WiFiAP device. Now AP sends the packets to clients using Weighted Fair Queue Scheduler.

### 4.0.1 Why Intermediate Queue at WiFiAP node?

It is very difficult to control clients in wireless network. Intermediate queue is only installed on AP node because it is easy to control one node where all clients are connected. So, it supports only downlink traffic(from AP to clients).

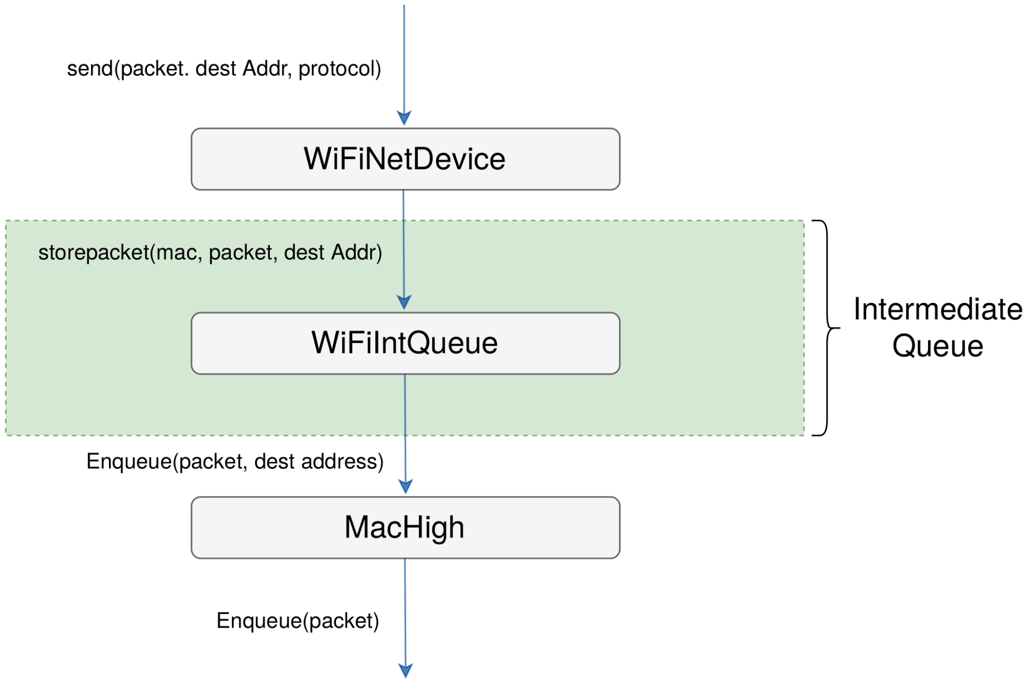



Figure 4.1: WiFiIntQueue at AP

There are two main functions in WiFiIntQueue class:
(i). storePacket(mac,packet,dest addr) and
(ii). callEnque()

### 4.0.2 storePacket(mac,packet,dest addr)

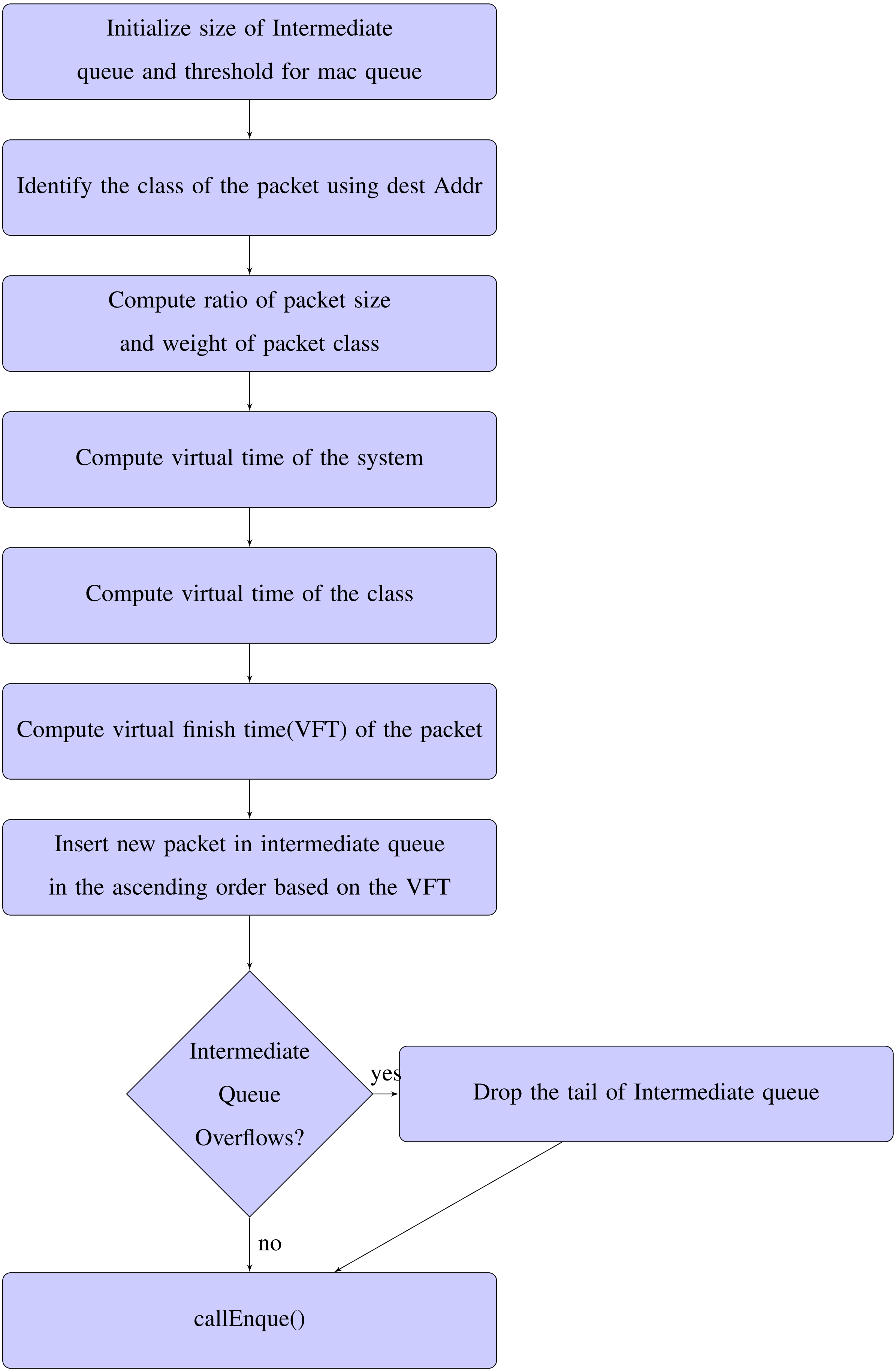
Whenever WiFiNetDevice sends packet to WiFiIntQueue, storePacket function is invoked first. In this function, the virtual finish time of packet is computed using equation 4.1

$$VFT\ of\ packet = \frac{packet - size}{weight\ of\ packet\ class} + max(\ finish\ time\ of\ system,\ finish\ time\ of\ class) \quad (4.1)$$

where,
**Finish time of system** is defined as virtual finish time of last packet which is sent to mac queue, and it is zero when intermediate queue is empty.
**Finish time of class** is defined as the largest virtual finish time of packet of that class in the queue(i.e. the last packet of that class in queue), and it is also zero when intermediate queue is empty.

Maximum size of Intermediate queue and threshold for mac queue is initialized first. Whenever a new packet arrives to the intermediate queue, its VFT is calculated using 4.1 and is tagged with its VFT. And then it is inserted into the queue based on ascending order of VFT of packets. After that size of the queue is checked. If it overflows, tail of the queue is dropped. Tail may be or may not be the current arrived packet, since it depends on its VFT. It is clearly mentioned in the flowchart on next page.

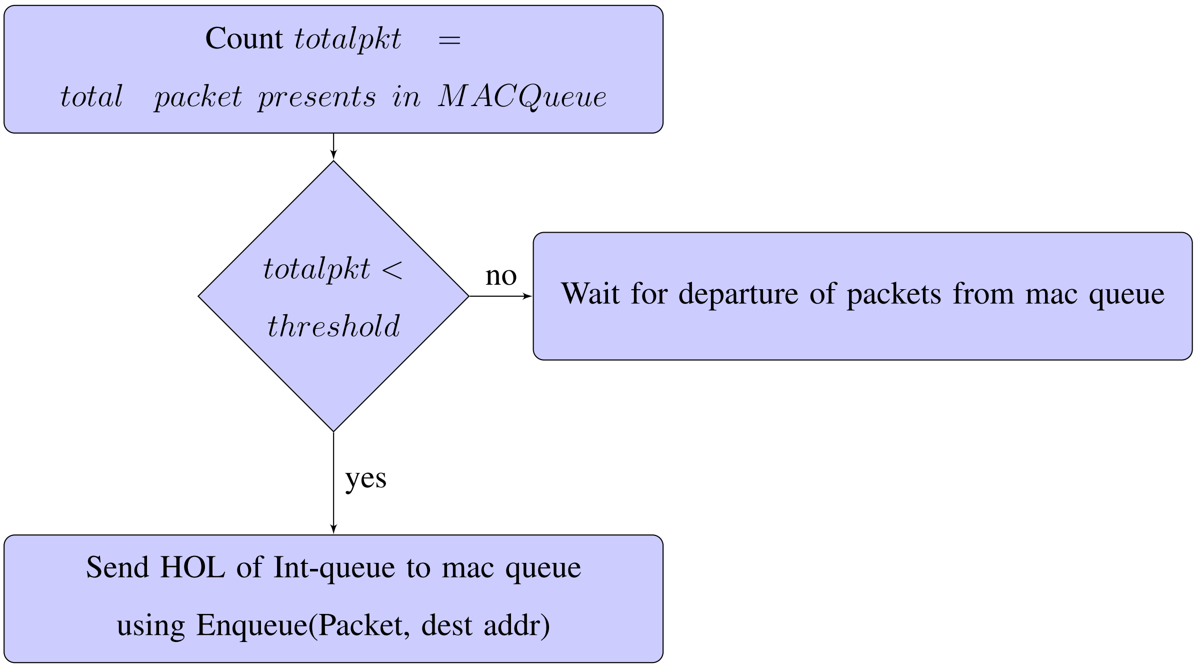
**Flowchart of storePacket()**



### 4.0.3 callEnque()

In WiFiIntQueue, callEnqueu() is invoked after storePacket(). This function checks total packets in mac queue before sending a packet. A threshold is set for this so that packets in mac queue never exceeds set threshold.

**Flowchart of callEnque()**



# CHAPTER 5

# SIMULATION RESULTS

## 5.1 Throughput Of Two Clients with WFQ

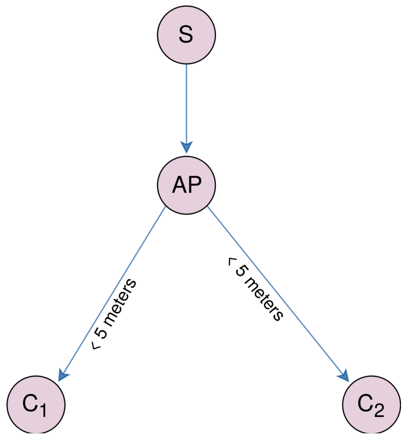



Figure 5.1: 1 Server 1 AP and 2 Clients for WFQ Experiment

### 5.1.1 UDP

There are 1 server, 1 AP and 2 clients are in the notwork and to check the performance of WFQ in UDP traffic, the weight of the client is changed. Detail simulation setup is mentioned in Table 5.1 and the throughput of both clients are shown in figure 5.2. Here the ratio of throughput of two clients is same as the ratio of the weight of those clients. So, the throughput of a client is controlled by controlling its weight.

Table 5.1: simulation setup of WFQ for UDP

| | |
|---|---|
| IEEE Standards | 802.11ac |
| Traffic | UDP Downlink |
| Rate Adaptation | ContantRate VHTMCS8 |
| Maximum size of Intermediate queue | 10 Packet |
| Threshold for MAC queue | 10 Packet |
| Simulation Time | 5 sec(15-20 sec) |

Table 5.2: UDP Downlink Throughput of Two Clients with different weights

| Client | Weight | Throughput(Mbps) | Total Bytes Received |
|---|---|---|---|
| Client1 | 1 | 31.0448 | 19403000 |
| Client2 | 1 | 30.9563 | 19309000 |
| Aggregate Throughput | 62.0011 Mbps | | |
| Client1 | 1 | 20.0944 | 12559000 |
| Client2 | 2 | 39.9743 | 24934000 |
| Aggregate Throughput | 60.0687 Mbps | | |
| Client1 | 2 | 40.0416 | 25026000 |
| Client2 | 1 | 19.9888 | 12468000 |
| Aggregate Throughput | 60.0304 Mbps | | |

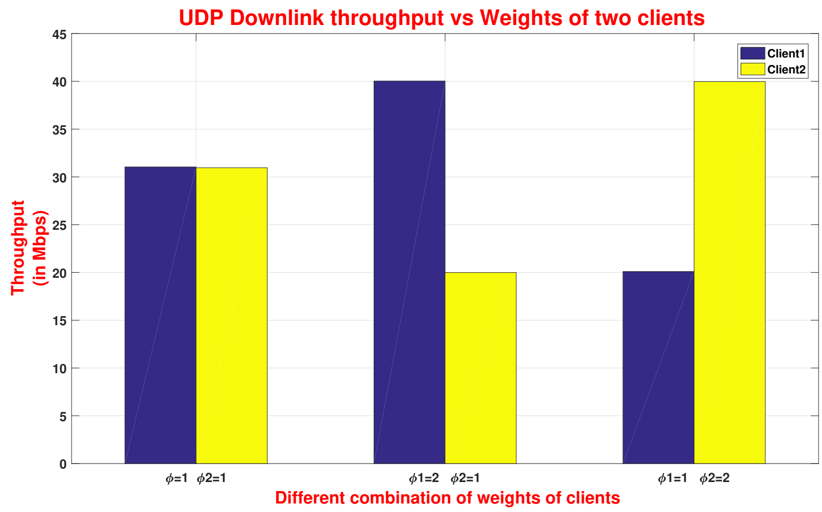



Figure 5.2: UDP Downlink throughput of two clients

### 5.1.2 TCP

To check the performance of WFQ in TCP traffic, the weight of the client is changed. Detail simulation setup is mentioned in Table 5.3 and the throughput of both clients are shown in figure 5.3. Here also the ratio of throughput of two clients is same as the ratio of the weight of those clients. So, the throughput of a client is controlled by controlling its weight.

Table 5.3: simulation setup of WFQ for TCP

| | |
|---|---|
| IEEE Standards | 802.11ac |
| Traffic | TCP Downlink |
| Rate Adaptation | ContantRate VHTMCS8 |
| Maximum size of Intermediate queue | 10 Packet |
| Threshold for MAC queue | 10 Packets |
| Simulation Time | 10 sec(15-25 sec) |

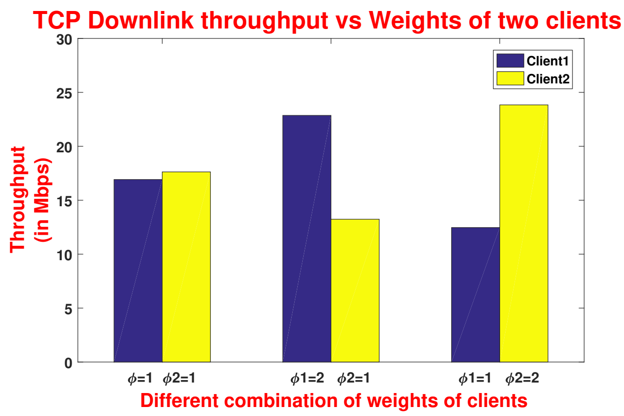



Figure 5.3: TCP Downlink throughput of two clients

Table 5.4: TCP Downlink Throughput of Two Clients with different weights

| Client | Weight | Throughput(Mbps) | Total Bytes Received |
|---|---|---|---|
| Client1<br>Client2 | 1<br>1 | 16.9213<br>17.6272 | 21151632<br>22031744 |
| Aggregate Throughput | 34.5485 Mbps | | |
| Client1<br>Client2 | 2<br>1 | 22.8653<br>13.2378 | 28581664<br>16545552 |
| Aggregate Throughput | 36.1031 Mbps | | |
| Client1<br>Client2 | 1<br>2 | 12.4664<br>23.8394 | 15582960<br>29796240 |
| Aggregate Throughput | 36.3057 Mbps | | |

## 5.2 Throughput Of Clients vs Packet Threshold of MAC Queue

In this simulation setup ( see Table 5.5 ), it shows that how throughput of two clients with equal weight are changing and how aggregate throughput is changing.

Table 5.5: simulation setup3

| | |
|---|---|
| IEEE Standards | 802.11ac |
| Traffic | UDP Downlink |
| Rate Adaptation | ContantRate VHTMCS8 |
| Maximum size of Intermediate queue | 10 Packet |
| Threshold for MAC queue | changing form 10 to 100 Packet |
| Simulation Time | 5 sec(15-20 sec) |

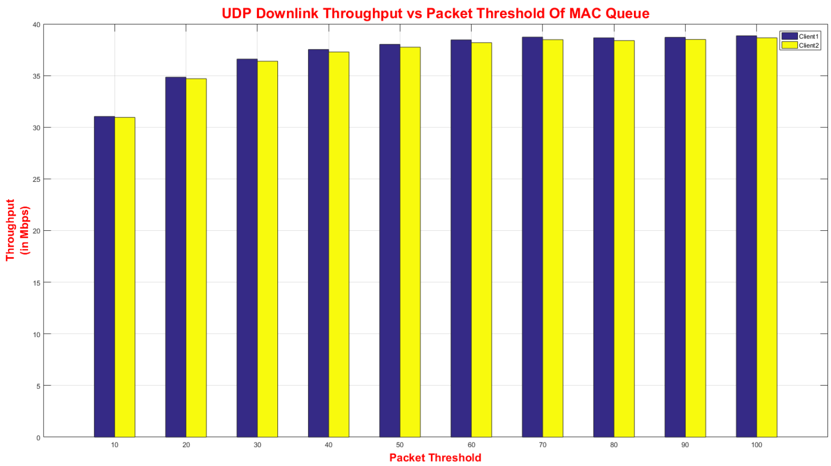



Figure 5.4: UDP Downlink throughput vs packet threshold of mac-queue

In Figure 5.4 shows that the throughputs of both clients are almost same, since they have equal weight. But aggregate throughput increases as threshold increases for UDP downlink traffic.

## 5.3 Log Distance Path Loss Model

Table 5.6: RSSI-MCS Mapping

| RSSI Range | MCS |
|---|---|
| -85.23 to -81.89 | 0 |
| -81.29 to -75.68 | 2 |
| -75.68 to -71.80 | 4 |
| -71.80 to -70.48 | 5 |
| -70.48 to -69.23 | 6 |
| -69.23 to -64.28 | 7 |
| -64.23 to 999999 | 8 |

if $d \geq d_0$

$$PL(d) = PL(d_0) + 10 * n * log(\frac{d}{d_0}) \tag{5.1}$$

else

$$PL(d) = PL(d_0) \tag{5.2}$$

where,

$PL(d) \triangleq Pathloss\ at\ distance\ d\ from\ AP$

$PL(d_0) \triangleq Pathloss\ at\ reference\ distance\ d_0$

$n \triangleq exponential\ pathloss\ model$

$$rssi = TxPower - PL(d) \tag{5.3}$$

Now, derive equation for distance of client from AP using equation 5.1

$$d = d_0 * 10^{\dfrac{TxPower - PL(d_0) - rssi}{10n}} \tag{5.4}$$

Table 5.7 shows the distance range of client from AP and this range is determined using Table 5.6 and equation 5.4 when $TxPower = 23dBm, \quad d_0 = 1m, \ PL(d_0) = 46.667dB \ and \ n = 3.5$.

Table 5.7: Distance-RSSI-MCS-Datarate Mapping

| RSSI Range(dB) | MCS | Distance Range(Meters) | Data Rate(Mbps) |
|---|---|---|---|
| -85.23 to -81.89 | 0 | 46.0478 to 57.3637 | 7.2 |
| -81.29 to -75.68 | 2 | 30.6041 to 46.0478 | 21.7 |
| -75.68 to -71.80 | 4 | 23.7095 to 30.6041 | 43.3 |
| -71.80 to -70.48 | 5 | 21.7374 to 23.7095 | 57.8 |
| -70.48 to -69.23 | 6 | 20.0214 to 21.7374 | 65 |
| -69.23 to -64.28 | 7 | 14.4566 to 20.1138 | 72.2 |
| -64.23 to 999999 | 8 | less than 14.4566 | 86.7 |

### 5.3.1 Fairness Among Clients : Minstrel

Now, the fairness is achieved for 5 clients those are using different data rate. To achieve it, Minstrel WiFiManager is used. It is a rate control algorithm. The following steps are followed to find the best data-rate of the client.

1. Find the distance between client and AP using Euclidean distance formula for all clients.
2. Find the rssi of each client using equation 5.3.
3. Find the best MCS for the rssi of client using Table 5.6.
4. Find the data rate [2] of a client for its MCS using Table 5.7.
5. Allot data-rate of the client as its weight.

Table 5.8: Simuation setup for 1 server 1 AP 5 Clients for both case1 and case2

| | |
|---|---|
| IEEE Standards | 802.11ac |
| Traffic | UDP Downlink |
| Rate Adaptation | Minstrel |
| TxPower of AP | 23 dBm |
| Channel Width | 20 MHz |
| SGI | Enabled |
| Maximum size of Intermediate queue | 10 Packet |
| Threshold for MAC queue | 10 Packet |
| Simulation Time | 5 sec(15-20 sec) |

**CASE 1 : When all clients are in same MCS range**

Table 5.9: Clients Details for Case1

| Client | Distance from AP(in meter) | RSSI(dB) | MCS | Data Rate(Mbps) |
|---|---|---|---|---|
| Client1 | 8 | -55.2858 | 8 | 86.7 |
| Client2 | 5.83095 | -50.4786 | 8 | 86.7 |
| Client3 | 10.4403 | -59.3327 | 8 | 86.7 |
| Client4 | 10.7703 | -59.8057 | 8 | 86.7 |
| Client5 | 10 | -58.6777 | 8 | 86.7 |

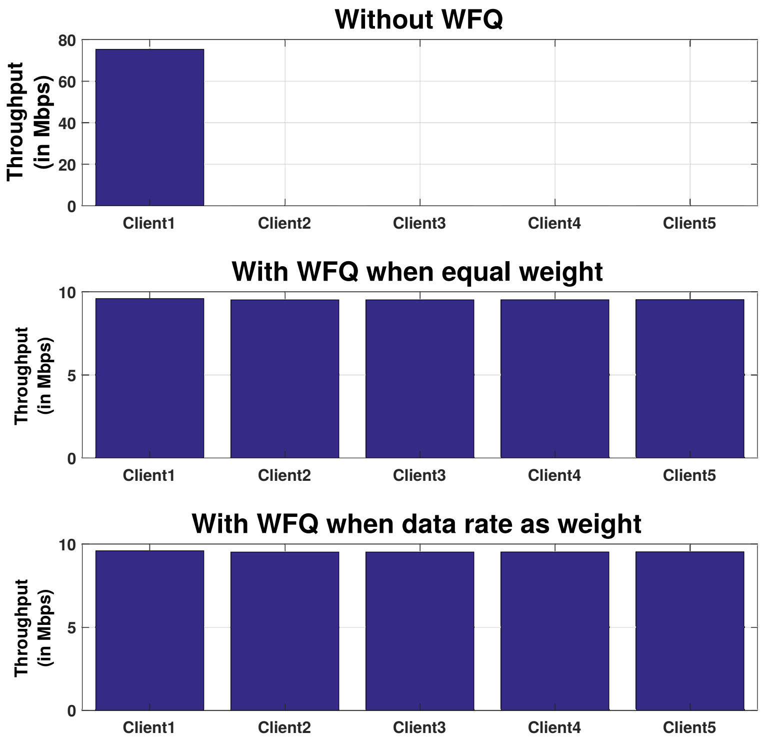



Figure 5.5: UDP Downlink throughput of 5 clients for Case 1

In Figure 5.5, the upper plot(without WFQ) shows that only one client is getting throughput, others are getting nothing. In middle plot WFQ is used for equal weight(1,1,1,1,1) for all clients and throughput of each clients is almost same. In lowest plot, WFQ with data rate is given as weight of client, here this plot is similar to middle one,since all have same data rate.

**CASE 2 : Except one all clients are in same MCS range**

Table 5.10: Clients Details for Case2

| Client | Distance from AP(in meter) | RSSI(dB) | MCS | Data Rate(Mbps) |
|---|---|---|---|---|
| Client1 | 20.5 | -69.5891 | 6 | 65 |
| Client2 | 5.83095 | -50.4786 | 8 | 86.7 |
| Client3 | 10.4403 | -59.3327 | 8 | 86.7 |
| Client4 | 10.7703 | -59.8057 | 8 | 86.7 |
| Client5 | 10 | -58.6777 | 8 | 86.7 |

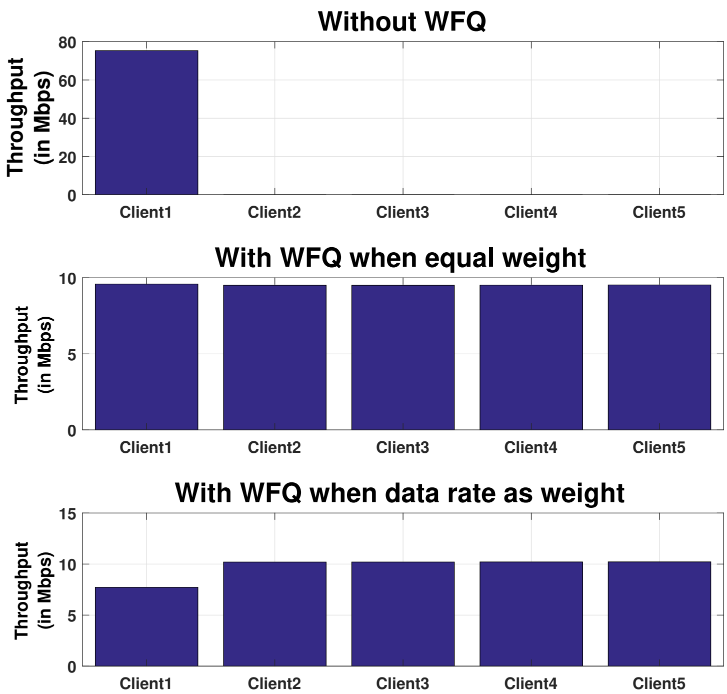



Figure 5.6: UDP Downlink throughput of 5 clients for Case 2

In Figure 5.6, the upper plot(without WFQ) shows that only one client is getting throughput, others are getting nothing. In middle plot WFQ is used for equal weight(1,1,1,1,1) for all clients and throughput of each clients is almost same. In lowest plot, WFQ with data rate is given as weight of client, the throughput of first client is lower than other, since its weight is 65 (its data rate for MCS 6) is smaller than others(86.7).

# CHAPTER 6

# CONCLUSION AND FUTURE WORKS

## 6.1 Conclusion

WFQ scheduler provides fairness to all clients for TCP and UDP downlink traffic and throughput of a client is control by varying its weight. The ratio of throughput of clients is same as the ratio of their weights. The ratio of throughput of client are not changing with set threshold for MAC queue but individual and aggregate throughput is affected. It provides fairness when the clients have different data rates.

## 6.2 Future Work

- In the future, this fair scheduler can be implemented for dense network scenarios where multiple access points(APs) and their clients will be present.
- AirTime fairness can also be implemented using this scheduler.
- IEEE 802.11ax standard can also implement scheduler in near future.

# CHAPTER 7

# REFERENCES


1. Network simulator ns-3, https://www.nsnam.org/
2. http://mcsindex.com/
3. **Toke, Kazior, Taht, huritg**(2017). Ending the Anomaly: Achieving Low Latency and Airtime Fairness in WiFi
4. **L. Jiang and S.C. Liew**(2005). Proportional Fairness in WLANS and Ad Hoc Networks.
5. **M.Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda**(2003). Performance Anomaly of 802.11b
6. **D. Bertsekas and R. Gallager**. Data Networks. Prentice Hall, second edition.1992
7. **G.Tan and J. Guttag**(2004). Time-based fairness improves performance in multi-rate WLANs