Control Theoretic Approaches for Efficient Transmission on IEEE 802.11e Wireless Networks

by

Ibukunoluwa Akinyemi

A Doctoral Thesis

Submitted in partial fulfilment
of the requirements for the award of

Doctor of Philosophy
of
Loughborough University

Abstract

With the increasing use of multimedia applications on the wireless network, the functionalities of the IEEE 802.11 WLAN was extended to allow traffic differentiation so that priority traffic gets quicker service time depending on their Quality of Service (QoS) requirements. The extended functionalities contained in the IEEE Medium Access Control (MAC) and Physical Layer (PHY) Specifications, i.e. the IEEE 802.11e specifications, are recommended values for channel access parameters along traffic lines and the channel access parameters are: the Minimum Contention Window CW_{min}, Maximum Contention Window CW_{max}, Arbitration inter-frame space number, (AIFSN) and the Transmission Opportunity (TXOP). These default Enhanced Distributed Channel Access (EDCA) contention values used by each traffic type in accessing the wireless medium are only recommended values which could be adjusted or changed based on the condition of number of associated nodes on the network. In particular, we focus on the Contention Window (CW) parameter and it has been shown in [11] that when the number of nodes on the network is small, a smaller value of CW_{min} should be used for channel access in order to avoid underutilization of channel time and when the number of associated nodes is large, a larger value of CW_{min} should be used in order to avoid large collisions and retransmissions on the network.

Fortunately, allowance was made for these default values to be adjusted or changed but the challenge has been in designing an algorithm that constantly and automatically tunes the CW_{min} value so that the Access Point (AP) gives out the right CW_{min} value to be used on the WLAN and this value should be derived based on the level of activity experienced on the network or predefined QoS constraints while considering the dynamic nature of the WLAN.

In this thesis, we propose the use of feedback based control and we design a controller for wireless medium access. The controller will give an output which will be the EDCA CW_{min} value to be used by contending stations/nodes in accessing the medium and this value will be based on current WLAN conditions. We propose the use of feedback control due to its established mathematical concepts particularly for single-input-single-output systems and multi-variable systems which are scenarios

that apply to the WLAN.
The contributions made in this thesis are as follows:

- First, we apply the concept on a network with single traffic type and design a feedback controller that can be tuned based on the desired settling time T_S and the percentage overshoot. We apply the control algorithm by controlling throughput directly on the network and then we set the collision probability as the control variable. We analyse the functionalities of the controller in these two scenarios.

- Next, we apply the feedback control concept to a multiple access category (AC) network by defining some delay constraints and fairness criteria to be maintained on the network. We used the decentralised control approach with a decoupler controller.

- Finally, we designed centralised optimal controller using the linear quadratic method for the controller design. The effectiveness of the use of the decentralised and centralised controllers was also assessed. We evaluated the functionalities of the controllers using the Matlab/Simulink tool and did a stability analysis to show the range of stability of the controller.

  The comparisons and analysis provide valuable insight and showed that the concept of feedback control is comparable to the EDCA protocol and can be used in implementing QoS constraints on the WLAN.

Acknowledgements

I will like to express sincere gratitude to God Almighty for life, energy and giving me the ability to undertake this research. I will also like to thank my supervisor Professor Shuang-Hua Yang for support, guidance and advice throughout my time as a student within the department. I will particularly like to say thank you for your great patience particularly in the writing-up year when everything really went slowly. Next, I am really grateful to Xiaomin Chen for her invaluable support and help and being there to talk to about my research.

I will also like to appreciate all the people I got to meet within the department of Computer Science, Loughborough University, both academic, secretarial and colleagues - Judith Poulton, Christine Bagley, Nebrase, Noura. Thank you for the time together. To my brothers and sisters in Birmingham that are always so concerned about my finishing and finishing well, I say a big thank you for your thoughts and prayers.

I say a big thank you to my parents. I really would not have been able to accomplish this Thesis without your support and constant encouragements. I also remember my siblings for their constant encouragement and prayers. Thank you so much.

And finally, I will like to thank my husband for his continuing support, encouragements, being there always when I feel so frustrated and inadequate. Thank you so much and God bless you all

Contents

List of Figures

List of Tables

Abbreviations and Acronyms

AC Access Category
ACK - Acknowledgement
AIFS - Arbitration Inter-Frame space
AIFSN - Arbitration Inter-Frame space number
AP Access Point
CAP Contention Access Phase
CFP Contention Free Period
CP Contention Period
CSMA/CA Carrier Sense Multiple Access with Collision Avoidance
CTS Clear to Send
CW Contention Window
DCF - Distributed Coordination Function
DIFS - Distributed (coordination function) Inter-Frame Space
EDCA - Enhanced Distributed Channel Access
EDCAF - Enhanced Distributed Channel Access Function
EIFS - Extended Inter-Frame Space
HC - Hybrid Coordinator
HCCA - HCF Controlled Channel Access
HCF - Hybrid Coordination Function
IBSS - Independent Basic Service Set
IEEE Institute of Electrical and Electronic Engineers
IFS Inter-Frame Space
LAN - Local Area Network
MAC - Medium Access Control
NAV - Network Allocation Vector
P Proportional control
PC - Point Coordinator
PCF - Point Coordination Function
PHY - Physical Layer
PI Proportional and Integral controller
PID Proportional, Integral and Derivative controller

PIFS - Point (coordination function) Inter-Frame Space
QoS - Quality of Service
QSTA QoS aware STA
RTS - Request to Send
RX - receive or receiver
SIFS - Short Inter-Frame Space
SSRC - Station Short Retry Count
SLRC - Station Long Retry Count
STA - station
TX - transmit or transmitter
TXOP - transmission opportunity
WLAN - Wireless Local Area Network

Chapter 1

Introduction

The IEEE 802.11 wireless network technology is the most widely deployed communication technology for wireless computer networks. It is a dynamic technology that can be easily deployed and specifies channel access standards at the MAC and Physical layers of the communication stack. The legacy version of the standard defines two medium access functions - the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF). DCF is the main channel access method, it is contention based and uses the carrier sense multiple access with collision avoidance (CSMA/CA) mechanism for channel access but there are no quality of service (QoS) guarantees.

In order to incorporate acceptable Quality of Service (QoS) on the WLAN, the Institute of Electrical and Electronic Engineers (IEEE) working group 802.11e introduced the Hybrid Coordination Function (HCF) [62], a medium access protocol at the MAC layer. HCF implements QoS by introducing traffic differentiation among the different traffic types on the wireless network thereby making it possible to incorporate service differentiated QoS along the lines of the different classes of traffic as specified in the IEEE 802.11e draft. As a result, it was then possible to ensure that time and delay sensitive traffic have quicker access to the wireless medium in order to reduce latency, improve on jitters, reduce packet loss to an acceptable level or better than acceptable, while improving on throughput.

Defined in the HCF is the Enhanced Distributed Coordination Function (EDCF) which is the protocol of particular interest in this research work. The EDCF is the enhanced version of DCF and is the primary access method at the MAC layer as defined in the IEEE 802.11e standards. It is contention based with different traffic classes contending for channel access using different Enhanced Distributed Coordination Access (EDCA) channel access contention parameters. The parameters as defined in the draft are along traffic class lines with recommendations made for their values while still giving allowance for the values to be adjusted as suited to different network designers and network scenarios.

Implementing a WLAN with the recommended static EDCA parameters successfully prioritised delay sensitive traffic thereby implementing QoS on wireless networks. This also gave rise to some performance improvement but the aspect of adapting and changing the contention parameters based on network condition of unpredictable number of associated nodes and network load, which will further enhance performance of the EDCA protocol and reduce congestion, delays and packet loss is not usually implemented due to the difficulty of assessing wireless network conditions which can be very dynamic. The inability to tune these parameters produces a rather sub-optimal network [54, 58].

In literature, [23, 48] there has been research in the area of improving on EDCA Channel accessing parameters and how they can be dynamically tuned in order to optimize the network and improve data throughput but these methods are heuristic and are not particularly based on mathematical foundations.

In this thesis, instead of heuristic approach to developing a system of tuning the contention window values used on the wireless network, we develop an algorithm that is based on the mathematical theory of feedback control. This will enable the WLAN to dynamically tune the minimum contention window value to be used by contending nodes in accessing the wireless network. The tuning will be based on the number of nodes transmitting on the network. The algorithm also maintains predefined network conditions and has the additional advantage in that while there will be some additional overhead and computational costs, there will be no need to add new hardware to existing ones.

1.1 Contributions

This thesis is about a dynamic approach in determining the right contention window (CW) value to be used by stations or nodes on the IEEE 802.11e wireless network during contention as nodes attempt to access the channel. In order to achieve this dynamic approach, we use the established principles of feedback control which is mainly used in system, process and control engineering, and implement on the wireless network for the purpose of regulation, reference tracking and parameter adaptation. We implement this using different network scenario.

First, we determine the appropriate input and output pairing of the wireless network. We choose the contention window parameter as the input to the system. We paired this initially with throughput as the system output and then with the probability of collision as the system output. We considered the situation where we control the throughput obtainable on the wireless network directly based on the established relationship between throughput and contention window value [26]. Following the results from this, we also considered the situation where throughput

is not controlled directly but indirectly by controlling the collision probability on the WLAN. For both of these scenarios, we used WLAN with only one traffic type.

Secondly, we followed up based on the conclusions of the first tests and we considered a multi traffic/multi access category (AC) wireless network since this depicts the realistic scenario of what is obtainable on most networks. This leads to a multi-variable control situation with multiple-input-multiple-output to the network with interactions between the inputs. To resolve interactions, we use the decentralised, decoupling method of control and was able to control each traffic type independently.

After decoupling, we considered a scenario where the range of CW_{min} values that can be given to the wireless system is constrained within a pre-defined limit. This introduced some level of saturation into the system and anti-windup controllers were designed to counter the saturation effects.

Finally, we designed a centralised controller using the Linear Quadratic Integral (LQI) control techniques. This involves the description of the wireless network using linear differential equations and a quadratic cost function whose solution is given in the design of a centralised controller that uses an optimal feedback law to minimize the value of the cost function.

1.2 Thesis Structure

This report is structured as follows:

In chapter 2, we gave an overview of the IEEE 802.11e wireless network and an overview on the theory of the Proportional-Integral (PI) controller. We explained the evolution of the IEEE 802.11e EDCA protocol, its mode of channel access and gave a description of the different contention parameters used in accessing the WLAN and their functionalities. We also discussed feedback control theory, how it functions and the two components of the PI controller. We concluded with related works.

In chapter 3, we extended the PI controller in [54] and implemented on an unsaturated network using IEEE 802.11a configurations. This chapter shows the implementation of the $SISO$ controller algorithm on a network where the station throughput and probability of collision were chosen as controlled variables. In chapter 4, we proceed to multi-variable control and the $MIMO$ WLAN. We implemented the control algorithm in chapter 3 with a decoupler. In chapter 5 we looked at an erroneous WLAN and implemented the control algorithm with system input constraints. In chapter 6 we used an optimal controller for the same $MIMO$ system and compared the effect of the decoupler and the linear quadratic controller on the $MIMO$ system and in chapter 7 we concluded this report and

discussed future works.

1.3 List of Publications

Journal

- Xiaomin Chen, Ibukunoluwa Akinyemi and Shuang-Hua Yang: A Control Theoretic Approach to Achieve Proportional Fairness in 802.11e EDCA WLANs (Published: Computer Communications, Volume 75, 1 February 2016, pages 39-49)
- Ibukunoluwa Akinyemi, Shuang-Hua Yang: Feedback control algorithm for optimal throughput in IEEE 802.11e EDCA networks (Published: Systems Science and Control Engineering, Volume 5, Issue 1 July 2017, pages 321-330)

Conference

- Ibukunoluwa Akinyemi, Shuang-Hua Yang: Feedback based control for efficient transmission over IEEE 802.11e wireless networks (CACSUK: 22nd International Conference on Automation and Computing ICAC 2016)

Chapter 2

Channel Access and Control Theory Concepts

The IEEE 802.11 wireless technology is the most widely used technology for wireless connection of nodes and is usually used for Internet connections as well. The first version of the IEEE 802.11 standard consists of two modes of medium access - the contention based access coordinated by the Distributed Coordinated Function (DCF) protocol and the polled access coordinated by the Point Coordination Function (PCF). The contention mode of accessing the WLAN is the fundamental form of media access while polling occurs when the WLAN management device gives priority and right to transmit to stations with backlogged frames, making access to the medium contention free. The polling mode of access is an optional mode of access.

The DCF is the fundamental access method for the IEEE 802.11 network and its mode of operation was enhanced in order to incorporate prioritised Quality of Service (QoS). This enhancement gave rise to the Hybrid Coordination Function (HCF) which comprises of the HCF Controlled Access (HCCA) protocol for polled access and the Enhanced Distributed Channel Access (EDCA) mechanism for contention based access to the wireless medium. HCF prioritises access to the WLAN by mapping frames into four different Access Categories (ACs) - Voice, Video, Best Effort and Background. This operates such that each station can have a maximum of four (4) independent queues or Channel Access Functions (CAFs) and each queue can initiate a backoff process using different values of contention parameters when it has a frame to transmit.

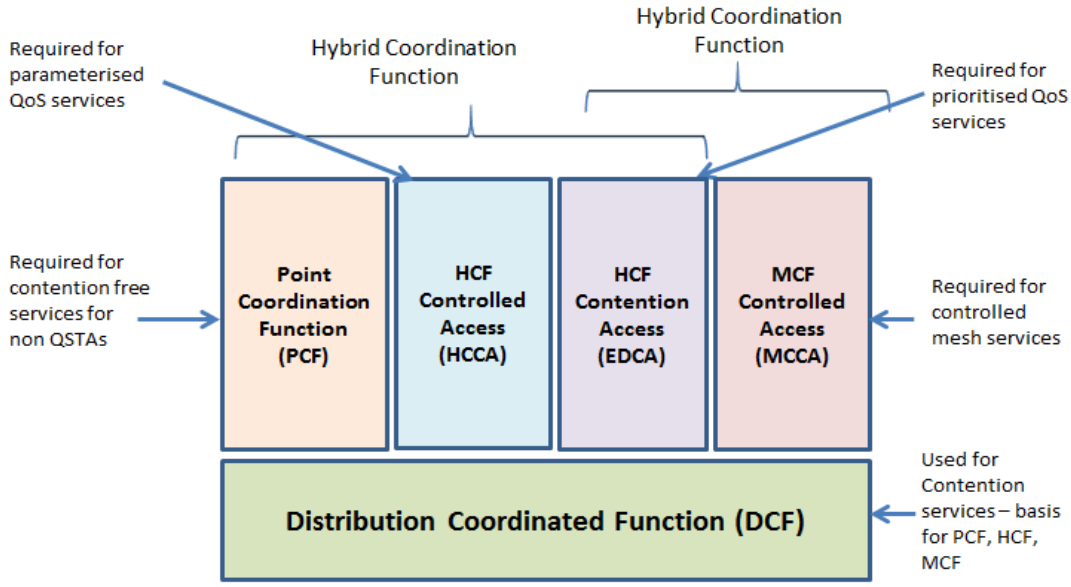

Figure 2.1: MAC Architecture [62]

2.1 EDCA Protocol

In a wireless network with more than one AC traversing the network, it is important to note that the QoS requirements of each traffic type varies and provision is made for this in the EDCA protocol. The EDCA implements the DCF medium access parameters - $Inter - frameSpacing$ (IFS), CW_{min}, CW_{max} and also the newly added Transmission Opportunity (TXOP) limit. In the following section, we will describe the EDCA contention parameters followed by a description of the EDCA channel access mechanism.

2.1.1 EDCA Access Parameters

The EDCA protocol differentiates between traffic on the wireless network and gives priority to delay sensitive traffic. This prioritization is achieved by assigning different values to the contention parameters used on the WLAN and they are defined as follows:

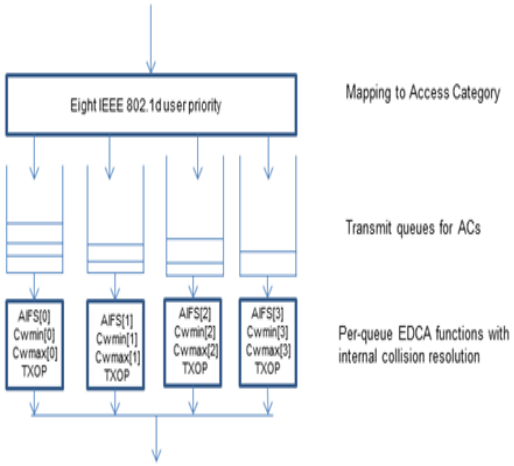

Figure 2.2: EDCA traffic Categories

Inter-frame Spacing (IFS)

Inter-frame Spacings are number of slot (time division on the WLAN) time stations observe in order to ensure the medium is idle before initiating a backoff.

The different types of IFS on the WLAN are:

- SIFS - This is the shortest IFS duration (T_{SIFS}). It is used when a station already has access to the medium and has to keep holding on in order to transmit all frames it has within its time allocation. SIFS time is given as the summation of some delay parameters and the maximum time in microseconds that it takes for a node to switch from receiving to transmission mode.

- AIFS - this is the time a station has to wait after sensing the medium as idle. It is a function of the physical layer slot time and higher priority data are assigned smaller AIFS value so that the time interval to wait before accessing the medium is smaller. The value of the AIFS time is derived from the equation:

$$AIFS[ACi] = AIFSN[ACi] \times aSlot\ time + T_{SIFS}$$

where $AIFSN[ACi]$ is an integer that shows the number of slot time a station must wait before initiating backoff. It has a range of 0 to 15, but the minimum value to be used for EDCA transmission is 2 because 1 is reserved for the access point. The value used per transmission is contained in the

dot11QAPEDCATableAIFSN field of the AP. An AIFSN value of 2 equals a DIFS.

- EIFS - the EIFS is the time duration a station has to wait following collision on the channel. It is defined as:

$$T_{EIFS} = T_{SIFS} + T_{DIFS} + T_{ACKTx}$$

where T_{ACKTx} is the time it takes to transmit an acknowledgement frame at the lowest physical layer mandatory rate [62].

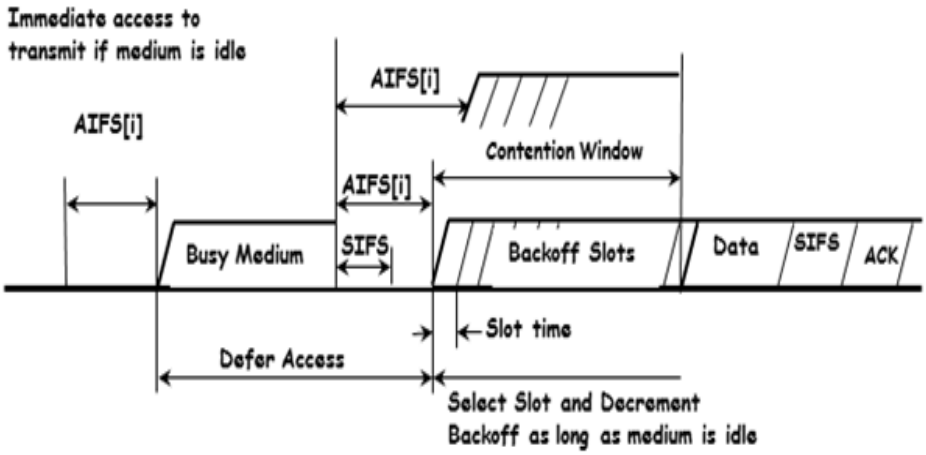

Figure 2.3: EDCF Inter-frame space relationships [62]

Contention Window

The EDCA protocol is a carrier sense protocol that ensures a wireless medium is idle before a node can initiate transmission of frames on the network thus minimizing the probability of collision. EDCA's method of achieving this is by implementing the backoff algorithm which is based on the contention window parameter. The backoff algorithm, involves the selection of a random integer from an interval $[0, CW_i - 1]$. The integer value chosen by a node signifies the number of consecutive empty slots it has to count down before attempting transmission. The minimum contention window $CW_{i,min}$ value is the initial upper limit of the CW interval. After the random backoff value is picked by a station, its countdown timer is set and during the countdown period, the station uses its carrier sense mechanism to detect a busy or an idle slot time. If the slot is idle, the backoff timer

will count down by the end of the slot time. If during the backoff count down, a slot is sensed as busy, the station freezes its timer and resume countdown only when the medium is sensed as idle again after an $AIFS_i$ period [11,18,56]. After the backoff period, the station will attempt transmission of its frame. If collision occurs during this transmission attempt, the station attempt retransmission by going through the backoff process again but this time, it will reset the CW parameter to a new value. The computation of the CW value to be used during retransmission is given by:

-

$$CW_i = 2^j CW_{i,min}$$

where j is the number of retransmission attempt and takes value from $[0, SSRC/or/SLRC]$. j is zero when the station makes an initial transmission which implies the station uses CW_{min} as initial value. The maximum contention window value is the value of CW when maximum retry limit is reached. If collision continues after the maximum retry limit is reached, a station either continues to attempt transmission using the CW_{max} value of the frame or the frame is dropped and CW value is set back to the minimum value. If a successful transmission occurs, the CW size is also re-set to the $CW_{i,min}$ value.

Transmission opportunity (TXOP)

The EDCA TXOP is the duration of time a Channel Access Function (CAF) is permitted to retain access the wireless medium. It was introduced by the IEEE 802.11e group as part of the measures to avoid channel capture by stations and is defined by a starting time and a maximum duration of use of the medium. There is the HCCA TXOP and the EDCA TXOP. The HCCA TXOP is obtained when a station is polled by the AP while the EDCA TXOP is obtained when a station obtains the right to access the medium and transmit its frames. The duration of the TXOP is set in the beacon frame as advertised by the AP and a value of zero implies within the current TXOP, only one frame can be sent with the required acknowledgement or the RTS/CTS frames depending on the mode of transfer. If the TXOP limit is set to a non-zero value, the station can send fragmented frames without exceeding the value of the TXOP limit.

Priority	Access Category [AC]	Service Type	AIFSN	CW_{min}	CW_{max}	Max TXOP
Lowest	AC [0]	Background	7	31	1023	0
↓	AC [1]	Best Effort	3	31	1023	0
↓	AC [2]	Video	2	15	31	3.008ms
Highest	AC [3]	Voice	2	7	15	1.504ms

Figure 2.4: Access categories and corresponding 802.11e values

2.1.2 Media Access

The IEEE 802.11e media access protocol is defined at the Medium Access Control (MAC) and Physical (PHY) layers of the network communication stack. EDCA operates a carrier sense multiple access with collision avoidance (CSMA/CA) mechanism. The aim of any channel access method is to ensure a successful packet transmission while avoiding collision on the WLAN. Two modes of transmission are defined within EDCA. These are the Basic Channel Access mode of transmission and the Request-to-send (RTS) Clear-to-Send (CTS) mode of transmission.

Basic Channel Access method

The basic access method is the fundamental contention medium access method that determines when a station gains access to the wireless medium. The diagram for the basic channel access method is shown in Fig 2.5. In the basic channel access method, when a station has a frame from the traffic category AC_i to transmit, it must be in listening mode to ensure no transmission is going on on the WLAN [46]. This is to avoid collision. After sensing the medium as idle for an $AIFS_i$ period, the station may then attempt transmission. If the medium is not idle, the station must wait and continue to monitor the channel until idle. After the $AIFS_i$ interval, the station performs a random backoff and starts its backoff timer to count down.

$$\text{Backoff}\left[AC_i\right] = Random\left[0, 2^j CW_{i,min} - 1\right] \tag{2.1}$$

For the backoff timer, the integer range to be used depends on the current contention window value of AC_i. During backoff, the counter decrements by one

at the end of every empty slot time if the medium is idle. When the backoff timer gets to zero, the station then transmits its packet and the transmission can either be a successful one or a collision occurs. If collision occurs, the station must wait an EIFS interval before re-attempting the backoff process and re-transmission.

If a collision occurred, the backoff integer range increases exponentially as shown in equation 2.1. The maximum value that CW_i can assume is stipulated in the standards and shown in Fig 2.4.

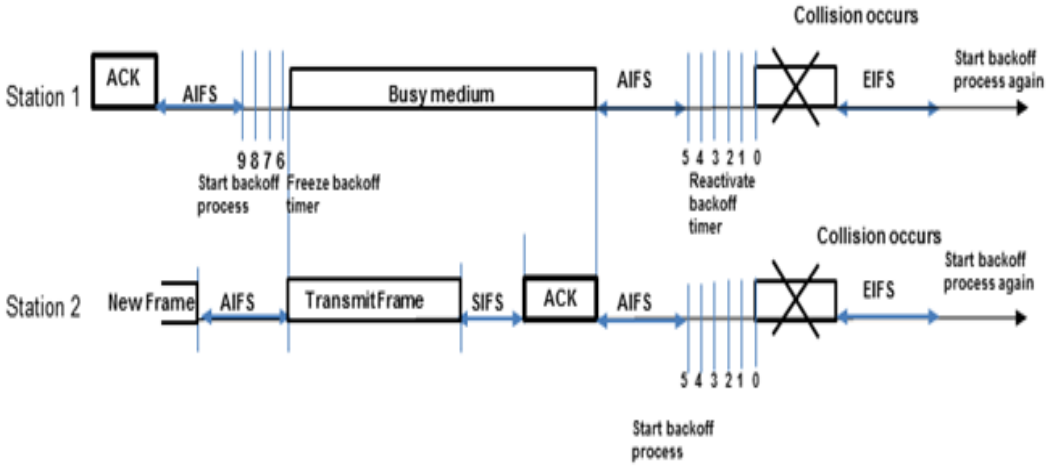

Figure 2.5: EDCF basic access procedure

At the end of every transmission, the receiving node must send an acknowledgement frame (ACK) to the transmitting node signifying receipt of the transmitted frame. Performance of the MAC layer protocol is analysed in relation to the QoS requirements of each traffic class and particularly for WLANs with heterogeneous traffic, the delay constraints for voice and video transmissions is critical and must be achieved for a good user experience. Achieving good performance also depends on the ability of the protocol-algorithm to scale well in the face of increasing number of stations or change in network scenario. These depend on different factors that are characteristics of the network such as channel accessing parameters, probability of collision and rate of retransmission on the WLAN and other forms of losses experienced on the network.

RTS/CTS Channel Access method

On the 802.11e WLAN, every node can sense the transmission of stations within its transmission range. In order to further avoid collision especially in the case of

hidden nodes, the four way handshake or RTS/CTS mode of initiating transmission was designed as a means of reserving the channel for transmission. In the RTS/CTS mode of transmission, the sender initiates communication by sending a request-to-send packet to the intended receiver after sensing the medium is idle and performing the backoff process. All nodes on the network within the transmission range of the sender should see this packet and know that a transmission is about to take place. If the receiver is able to communicate, it sends a clear-to-send packet back to the sender. All nodes on the network should receive either of the two handshake packets which will also contain the duration of the intended communication. Once the handshake is concluded, communication can begin and all other nodes set their Network Allocation Vector (NAV) timer accordingly.

The NAV is a timer/indicator maintained periods when the wireless medium will be busy with transmissions.

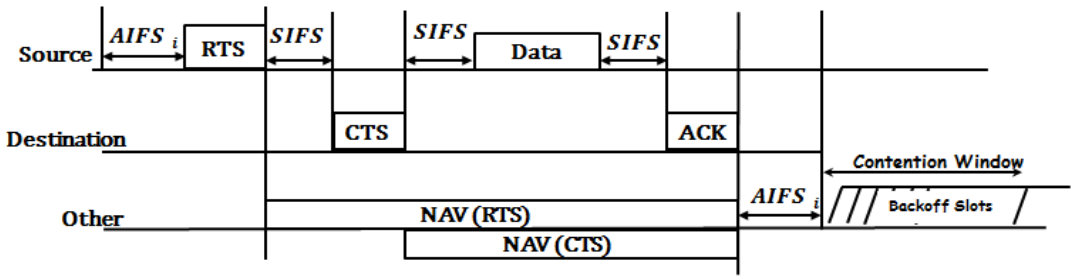

Figure 2.6: Four-way Handshake Channel Access

2.2 Control Theory

Feedback control deals with dynamic systems and how their behaviour can be controlled in order to achieve desired outputs while reducing the sensitivity of the system to disturbances [50, 61]. It involves modelling the system with respect to its input and output parameters and then designing appropriate feedback controllers that feed the system to be controlled with inputs that enable them function with respect to pre-defined tasks. It takes into account the current conditions of the system to be controlled including disturbances experienced by the system and adjust the dynamics of the system such that desired output is achieved. Feedback control theory can be used for different types of control tasks such as regulation and disturbance rejection, reference tracking, adaptation of tunable parameters

etc [6, 25, 31, 50, 51]. This research work presents the use of the concept of feedback control for the purpose of regulating and tuning of network parameters particularly in the face of disturbances such as the dynamic nature and movement of nodes on a network. A feedback controller is designed that ensures design condition is maintained on the network without loss of system stability. The target or design level to be maintained is called the reference and the system output that is measured to ensure the target is achieved is called the controlled variable as shown in Fig 2.7. The system output and the reference variable are of the same unit.

In general, control systems have two configurations: Open loop control systems and closed loop or feedback control systems. We use the closed loop architecture in this work. Particularly, we will design a closed loop system that takes the throughput value or the probability of collision value on the wireless LAN, measures it against the reference value and the designed controller will use the measured error to compute channel accessing parameters.

Feedback control system operates broadly in four steps

- Step 1: It measures the system output (y)
- Step 2: The measured output is fed back and compared to a predefined reference at the summing junction
- Step 3: The error margin $e(t)$ is computed
- Step 4: System input parameter is manipulated by the controller and corrected using the error margin so that predefined output is obtained.

Control systems can have different number of inputs and outputs to the controlled system. The number of input-output control variables are normally a form of classified for control systems with single-input single-output systems referred to as $SISO$ systems and multiple-input multiple-output systems referred to as $MIMO$ systems. Classical control theory concepts are usually used for $SISO$ systems while multi-variable control and modern control theory are used for $MIMO$ systems

Feedback control has previously been used in tuning network parameters in wired and wireless [13, 14, 16, 29, 54] networks, an example being in the Random Early Detection (RED) protocol. While feedback control systems have obvious advantages, it also has the potential of introducing instabilities to a system.

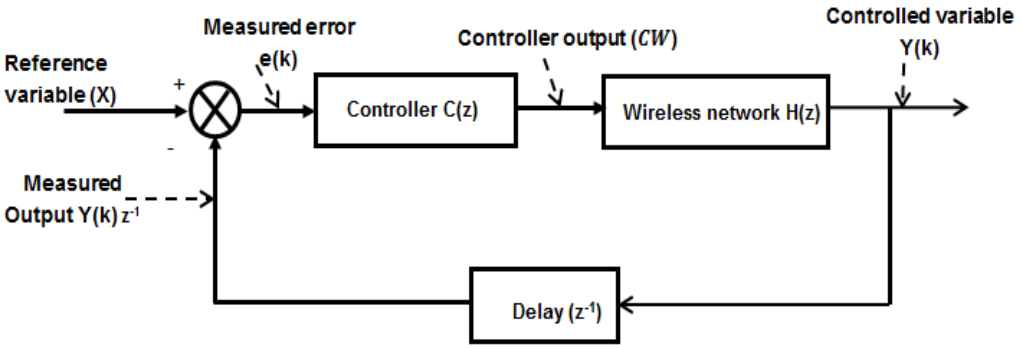

Figure 2.7: Feedback control block diagram [21]

2.2.1 Single-Input-Single-Output System

In designing a *SISO* control system, an orderly representation/layout of signal or process flow is necessary. This is depicted using block diagrams where the output of a subsystem is the input of another subsystem (see fig 2.7). With the block diagram, mathematical model showing the linear relationship between input and output of each subsystem can be derived. This mathematical model of the output-input relationship is called the transfer function of the system and to evaluate a system and obtain values for the controller parameters, the block diagram of inter-connected subsystems need to be reduced to a single block with a loop transfer function. In classical control, controller parameters are composed of three individual components - Proportional (P), Integral (I) and Derivative (D) controllers and each has different function and effect on the system based on the tuning parameters. These controller are perform better when combined depending on task to be achieved.

Proportional-Integral (PI) control

For the control of the wireless network, we used the PI controller. The PI controller consists of two tuning parameters - the Proportional (P) and the Integral (I) components and they aim at getting the error $e(t)$ to zero and doing that at the right speed.

Proportional Control - The proportional control component (K_p) controls speed of response of the controller and gives an output signal that is proportional to the measured error. The P-controller can be used to achieve a steady system

with reduction in the measured error value but the system output never gets to the value of the reference input.

Integral Control - With Integral control, the controller takes into account the history of the error over time, sums it up and uses this summation in its control action to achieve a zero steady state error but with some effect on transient time i.e. time taken before steady state is reached. If the speed of the integral controller is fast, the time taken to correct the error is faster but at the expense of stability and if the effect is slow, the controller is sluggish so some compromise is necessary in order to achieve stability.

PI Controller - For our wireless system, we consider the fact that the access point (AP) gives as output the CW_{min} value to be used for contention at every beacon interval. So, from [50] and [54], we have that

$$CW_{min}(k) = K_p e(k) + K_i \sum_{p=0}^{T_s} e(k) \tag{2.2}$$

where k represents discrete time instances when the CW_{min} value to be used on the network is given out and it corresponds to the sampling time interval of $T_s = 100ms$

From [54], we note that a more efficient method incorporating the error $e(k)$ over time is to use the $CW_{min}(k-1)$ of the previous cycle in our computation. So, equation 2.2 then can be written as

$$CW_{min}(k) = CW_{min}(k-1) + K_p e(k) + K_i - K_p e(k-1) \tag{2.3}$$

This yields a difference equation that can be transformed by applying the Z transform to give:

$$CW_{min}(z) = CW_{min}(z)z^{-1} + K_p e(z) + K_i - K_p e(z)z^{-1} \tag{2.4}$$

To arrive at the general representation for the controller configuration, we rearrange the transformed expression to arrive at the controller transfer function as

$$CW_{min}(z)[1 - z^{-1}] = e(z)[K_p + (K_i - K_p)z^{-1}]$$

$$\frac{CW_{min}(z)}{e(z)} = \frac{K_p + (K_i - K_p)z^{-1}}{1 - z^{-1}}$$

$$C(z) = \frac{CW_{min}(z)}{e(z)} = \frac{K_p z + K_i - K_p}{z - 1} \tag{2.5}$$

or

$$CW_{min}(z) = e(z)\left[K_p + K_i\frac{1}{z-1}\right] \tag{2.6}$$

where K_i is the integral tuning parameter and $C(z)$ represents the controller transfer function.
To calculate the closed-loop (input-output) transfer function $G(z)$ for the system represented in Fig 2.7, and simplify the system by reducing it to its block diagram representation. We represent the system output $Y(k)$ as:

$$Y(k) = e(k)C(z) \times H(z) \tag{2.7}$$

but

$$e(k) = X - Y(k)z^{-1} \tag{2.8}$$

where the notation z^{-1} denotes a one step delay in the system. From equation 2.7 and equation 2.8, we have that:

$$Y(k) = \left[X - Y(k)z^{-1}\right]C(z)H(z)$$
$$Y(k)\left[1 + C(z)H(z)z^{-1}\right] = XC(z)H(z)$$

$$\frac{Y(k)}{X} = \frac{C(z)H(z)}{1 + C(z)H(z)z^{-1}} \tag{2.9}$$

Wireless System - The wireless system is a discreet system and we represent the block diagram as shown in Figure 2.8.

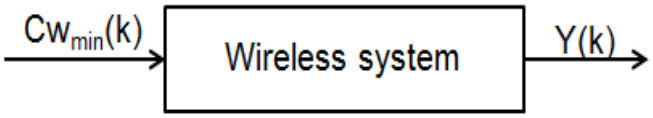

Figure 2.8: Wireless system block diagram

Applying the Z transform to the system gives a transfer function of

$$H(z) = \frac{Y(z)}{CW_{min}(z)} \tag{2.10}$$

The general representations in this section will be employed in this thesis.

2.2.2 Multi-variable Control

Multi-variable control involves the regulation and manipulation of multiple-input multiple output ($MIMO$) systems [2, 51, 61] in order to make it achieve certain pre-defined task. In multi-variable systems, the controller has to give more than one controller output that jointly control the system. Due to this joint control of the system and interconnection between subsystems, there are interactions between subsystems and this makes control of the system difficult.

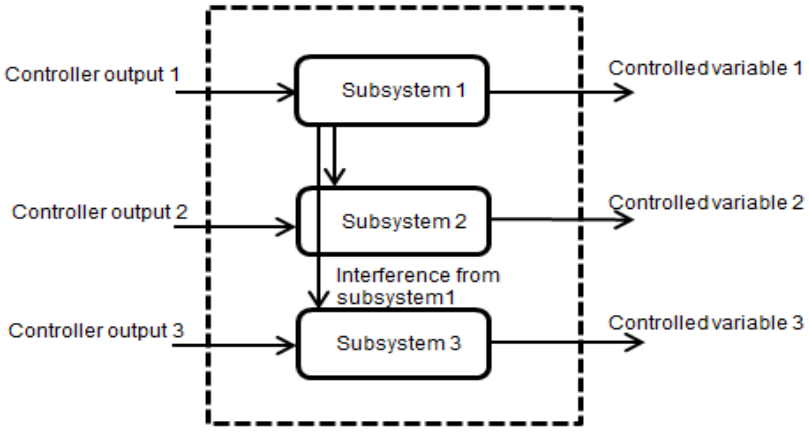

Figure 2.9: Multivariable Control System

To achieve the control of our $MIMO$ WLAN, we have looked at two approaches and compared their effectiveness. The approaches are:

- Decentralise and decoupling This is a two step approach that involves getting the system transfer function matrix G_s as close to diagonal as possible. A decoupling controller is used to counteract the interactions [5, 24, 51] if interactions between subsystems are not strong. When the system is decoupled, the subsystems can then be treated as $SISO$ systems. For this method to be used, there is need to check if the system can be decoupled in the first place by using the singular value decomposition (SVD) and condition

number (CN) to check the strength of interactions and then using the relative gain array (RGA) to select correct pairing of input-output variables.

- Linear optimal approach: this is a more modern control technique [2, 61] and it involves finding the optimal control law that operates a linear system at minimum cost using some weighing functions. A example of the control algorithm is shown in Figure 2.10.

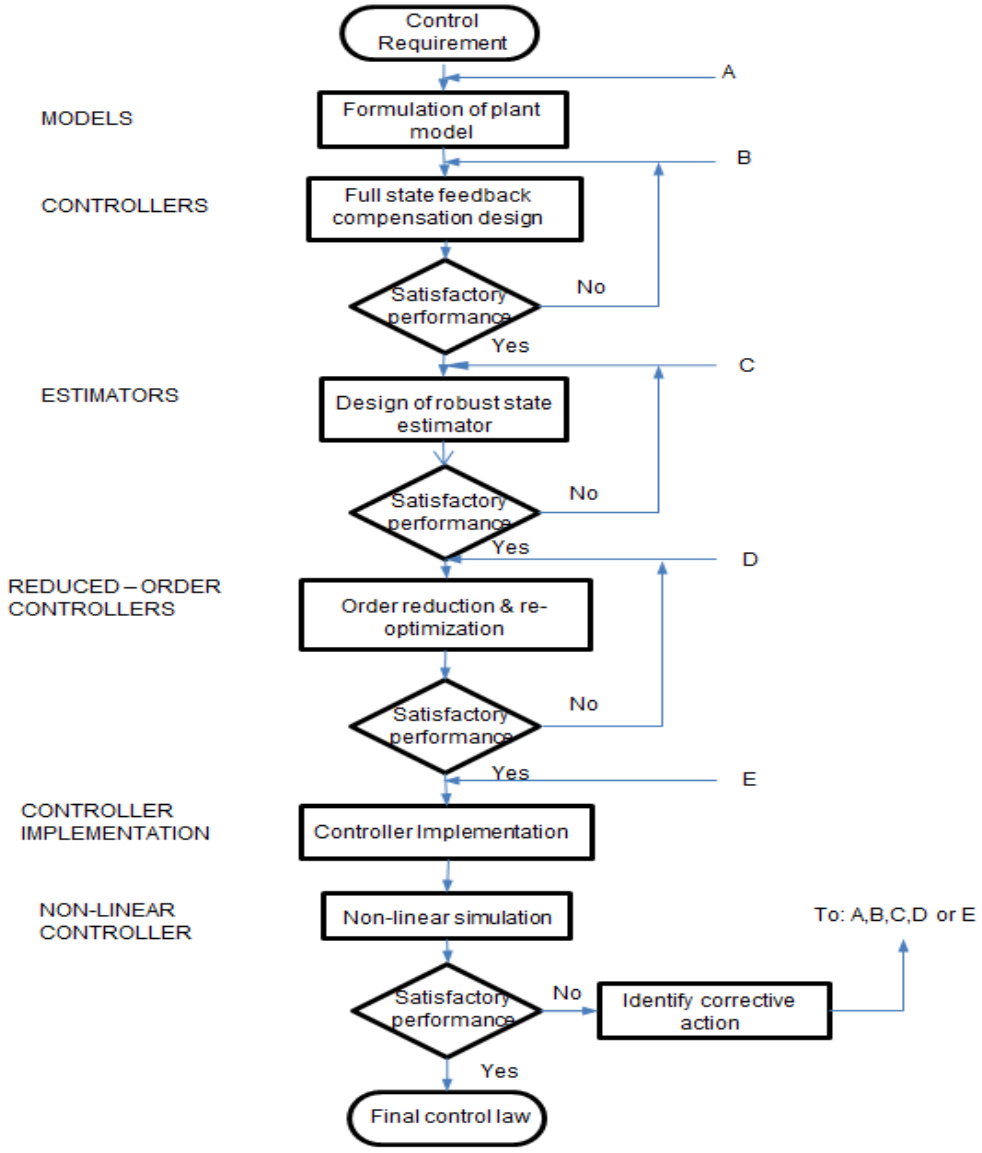

Figure 2.10: Control Law Synthesis Process [2]

2.3 Related Work

The EDCA protocol is a QoS protocol for traffic differentiation and prioritisation in heterogeneous WLAN. One of the methods used by the EDCA protocol to achieve traffic differentiation on the WLAN is by allocating different CW_{min} values to different traffic types and consequently, some amount of QoS is achieved but under saturation conditions or heavy load, EDCA cannot guarantee fair allocation of bandwidth and low priority traffic suffers [64]. Several proposals in literature have been made in order to maintain traffic differentiation while ensuring fairness in the sharing of network resources. In this section, we will discuss some related work with respect to tuning of the minimum contention window value, weighted and proportional fairness and the use of control theory in wireless networks.

Contention Window

Dynamic tuning of the contention window parameter (CW) has been a major source of consideration in order to improve on performance of the EDCA protocol. While the EDCA protocol provides QoS guarantees by prioritising delay sensitive traffic, performance on the WLAN can be further improved if the recommended CW_{min} values contained in the IEEE 802.11e standard [62] are dynamically tuned and adapted to network conditions instead of being statically set. This is because it has been shown through experiments in [26] that throughput on a WLAN depends largely on the number of contending stations and the value of $CWmin$. Also, the delay the head-of-line (HOL) packet experiences before accessing the network is largely affected by number of retransmissions occurring on the network which can be reduced if the right CW_{min} value is used on the WLAN. Adapting the CW values to network conditions has therefore been a subject of research.

In adapting contention window values to network scenarios, the challenge is in devising a means of assessing level of activity on the WLAN, in real time, and having a process of feeding this level of activity back to the WLAN management entity so that the right value of CW_{min} is broadcasted to contending nodes at each beacon interval. There are proposed schemes, most of which are concerned with mode of sensing the level of activity on the WLAN and adjusting the contention window value accordingly. Authors in [32, 49, 57] used the number of packets sent and the number of retransmissions within a specified period to calculated and judge the intensity of traffic on the WLAN and some multiplication factor is used to tune the CW values accordingly. Authors in [48] proposed a sliding contention window algorithm where a slider with a lower and upper limit value within the CWmin and CWmax values is used. Each traffic type uses contention window

values within a SCW slider range with upper and lower slider limits of the slides chosen within the specified IEEE 802.11e CW range. As the network gets busy, the slider range is adjusted by a sliding factor (SFi) which is AC specific. The challenge with most of these approaches shown above is the heuristic nature of the algorithm. In order to incorporate a theoretical approach to improve on this challenge, we use the concept of feedback control.

Fairness in WLANs

In a heterogeneous network, fairness is a form of measuring if users on a network receives a fair allocation of network resources [37]. A network being heterogeneous could be with respect to nodes on the network or in terms of the types of traffic/flows on the network and there are different types of fairness criteria that have been used for resource allocation in wireless networks. Some of the methods include time-based fairness, throughput-based fairness, max-min fairness [41] and proportional fairness [10, 22, 30, 35, 37, 42, 47] and some of the causes of unfairness in wireless networks are factors such as hidden terminals [30], exposed terminals, channel capture [40], uplink/downlink unfairness [33, 44, 63], asymmetric radio conditions, multiple data rates [8, 43, 60] and so on. In resolving fairness problems, different fairness measures are used but we have considered weighted and proportional fairness in heterogeneous WLAN.

In literature, there has been a number of fairness allocations that rely on assigning weights as a form of proportioning in order to share resources fairly on the network. Weighted fairness was used in bandwidth allocation for instance in [39, 64, 67]. In [39], a differentiated service EDCA (DS-EDCA) was proposed that supports both prioritisation and weighted fairness through the use of the AIFSN number. Network resource is first allocated to flows of higher priority and the remaining bandwidth is then shared among other flows according to allocated weights. Yuan et al [67] proposed iPAS which assigns priority to WLAN traffic in a dynamic way and consequently determines bandwidth allocation and [64] proposed a distributed weighted fair queuing (DWFQ) algorithm to ensure lower priority packets gets fair bandwidth allocation in a highly saturated network environment. Packet length and inter-arrival time is used to estimate throughput and different weights are applied to different flows such that the throughput to weight ratio for all flows are the same. In [4, 38, 44], weighted fairness was used in achieving fair channel access while in [33], the Channel Occupancy Time Based Rate Control for TCP (COTRC-TCP) was proposed as a method of resolving uplink/downlink unfairness

issues. This scheme controls the TCP window by monitoring both the number of active nodes on a network and channel occupancy at the MAC layer. In [40], a three dimensional makov chain model was proposed that determines how the backoff persistence factor is tuned so as to achieve a pre-determined weighted fairness among nodes in a wireless mesh network.

Apart from using weights to achieve fairness, authors have used the proportional fairness criterion based on the *log* of utility definition proposed in [34]. In [17, 20], authors characterised rate allocation using air-time and physical layer coding rate which allowed throughput,loss and delay trade-off among flows sharing network resources. [8, 60] both proposed algorithms that achieve fairness in a multi-rate IEEE 802.11e network such that stations are assigned a proportionally fair channel time irrespective of their transmission bit rate so that throughput achieved by a station is dependent on its sending rate. [60] included weights in their analysis. Other authors in [27, 43, 45, 52] also adopted the proportional fairness approach in allocating network resources.

Control Theory

Applying control theory concepts is not new in network communications as can be seen in [3, 12–15, 28, 29, 52, 65]. [29] analyses a combined TCP and Active Queue Management (AQM) model from a control theoretic standpoint. It uses a non-linear dynamic model of TCP to design a feedback control system depiction of AQM using the random early detection (RED) scheme. [15] introduces a control theoretical analysis of the closed-loop congestion control problem in packet networks. The control theoretical approach is used in a proportional rate controller, where packets are admitted into the network in accordance with network buffer occupancy. A Smith Predictor is used to deal with large propagation delays. [52] proposes a QoS-provisioning feedback control framework in order to achieve TCP uplink/downlink fairness and service differentiation. The medium access price (MAP) is delivered to TCP senders and the TCP senders adjust their sending rates to reduce congestion at the interface queue of the home gateway in an 802.11-based home network. Authors in [3, 12–14, 28] used the concept of feedback control in controlling bandwidth allocation to different traffic streams during the contention free period of the hybrid coordination function (HCF). Queue length of the different traffic type is fed back to the HCCA scheduler so that more appropriate bandwidth allocation is made to drain long traffic queues.

In [9, 53–55, 59], some work was done in improving throughput performance

on WLANs using contention based protocols for medium access. In [55], the Distributed Adaptive Control Algorithm (DAC) was introduced while in [59], the algorithm for implementing DAC was given. The approach was to design an algorithm that can be used on existing WLAN cards in a distributed WLAN i.e. a WLAN where the functionality to moderate medium access resides within each node implying the node computes its CW_{min} value for channel access. DAC implements a PI controller with the collision probability on the WLAN being the controlled variable. In [53,54,59], the centralised alternative to DAC, the Centralised Adaptive Control (CAC), was introduced. [53] introduces CAC algorithm for the dynamic control of CW_{min} parameter on the WLAN and [54] extends this algorithm by implementing on a WLAN with real-time traffic. The aim is to reduce delay and obtain a better Quality of Experience (QoE) for the video traffic. The concept of operation of both CAC and DAC are similar. While CAC resides in the access point (AP) and gives out CW_{min} value for contention to the nodes on the network, the DAC algorithm resides in individual nodes and both algorithms were tested on WLANs with single traffic type.

The work in this thesis extends the concept of the CAC algorithm in the following ways:

- In Chapter 3, we extend the CAC algorithm by incorporating the damping ratio ξ and natural frequency ω_n parameters for tuning the controller. This ensures a more robust controller for the single traffic network.

- In chapter 4, we considered an ideal WLAN with multiple traffic types traversing the network and we designed a decentralised control system which then enables the control of individual traffic type.

- Chapter 5 extends the functionality of the WLAN in chapter 4 by incorporating packet error into the modelling of the WLAN in order to depict a system that is as close to reality as possible. This chapter also extends the controller design and incorporates controller output saturation.

- Finally, in Chapter 6, we considered a centralised control system for a WLAN with multiple network traffic type.

2.4 Conclusion

In this chapter, we have looked at the evolution of the EDCA protocol, how it functions and we also have introduced classical control. For our research, we design for the PI controller based on the fact that of the three commonly used configurations of the controllers, the PI controller achieves the aim of automatically

driving the collision probability on the WLAN to the optimum value based on the condition of the WLAN while maintaining a stable system and without change in the existing infrastructures of the WLAN as defined in the IEEE 802.11e standards.

Chapter 3

Classical Control of a Single AC network

In this chapter we present the feedback control algorithm. This algorithm was developed using the principles of feedback control and has been applied to a WLAN transmitting single traffic type in order to maintain optimal conditions on the network while being able to tune the controller based on design specifications. Optimal condition on the network is monitored in two different ways.

In section 3.1, we use throughput as the control variable. Optimal condition on the network is achieved by monitoring the throughput achievable on the WLAN. The throughput on the network is tracked, compared with a pre-defined optimal value and the contention window CW_{min} value that achieves the desired throughput on the WLAN is broadcast to contending stations.

In section 3.2, we set the probability of collision on the network as the control variable and measure the network performance.

3.1 Throughput Control

To achieve this control task, the control layout used has four major components that are defined below:

- Reference Value S^{opt}: this is the desired throughput to be maintained on the WLAN. It is obtained based on physical layer technology used and number of contending stations as depicted in [11, 26].
- Observed network throughput S^{obs}: this is the throughput observed on the WLAN. It is the system output that is measured and compared with the reference/desired value until difference between them becomes zero.

- Controller $C(z)$: this is the mathematical representation of the designed controller. It is a transfer function of the difference between S^{opt} and S^{obs} and the system input.

- System representation $H(z)$: the system is modelled as a transfer function between the manipulated variable/system input, CW_{min}, and the controlled variable/system output, S^{obs}. Since these two parameters are non-linear as will be shown, we will linearise in order to obtain the required transfer function.

The mathematical models for the S^{opt} and the $H(z)$ layout components are derived while the controller representation is the pre-defined classical mathematical model of the PI controller. To begin the analysis, the throughput and delay expression for the network is first derived. We then proceed with the analysis of the performance of the algorithm with respect to network characteristics and the controller performance characteristics of peak time, settling time, steady state error and stability.

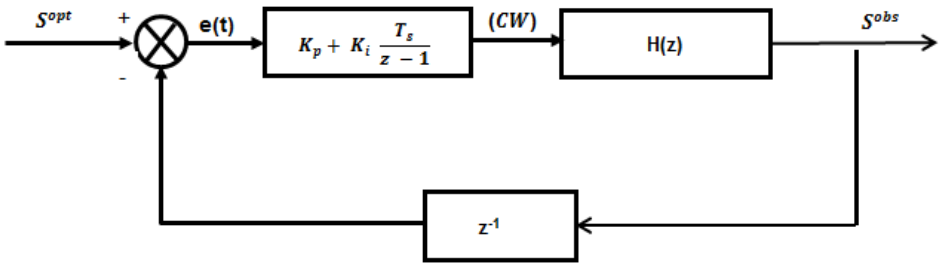

Figure 3.1: Control Layout

3.1.1 Throughput Analysis

This section presents the analytical model used for the throughput obtainable on the WLAN. For a network with n number of stations, the system throughput S is defined as the average rate of successful transmission of frames across the network.

This is represented as:

$$S = \frac{probability\ of\ successful\ transmission \times length\ of\ transmitted\ frame}{Expected\ duration\ of\ a\ contention\ slot}$$

With this definition, we note that a contention slot can either be empty or busy so we define the duration of a contention slot (T^{cs}) as the sum of the expected

duration of these states.

The probability of having an empty slot, P^e is the probability that none of the n stations on the WLAN attempts transmission. This probability is given as:

$$P^e = (1-\tau)^n \tag{3.1}$$

where the parameter τ, which denotes the probability that a station transmits a packet in a random slot time has been derived following the model in [11] and [36].

The probability that a slot time is busy,P^b, is defined as the probability that at least one of the n stations on the WLAN transmits in a random time slot. This is represented as

$$P^b = 1-(1-\tau)^n \tag{3.2}$$

A busy channel could either contain a successful transmission or a collision. A transmission is successful if only a single node out of n possible nodes transmits thereby avoiding collision on the WLAN. Thus, the probability of successful transmission of a single station is given as:

$$P^s = \tau(1-\tau)^{n-1} \tag{3.3}$$

and the probability of a collision in a slot time is given as

$$P^c = 1-(1-\tau)^n - n\tau(1-\tau)^{n-1} \tag{3.4}$$

To derive an expression for τ, we assume an unsaturated network scenario where traffic to be transmitted arrive in a Poisson distribution [36] and without considering exponential backoff meaning we set the number of retransmission on the WLAN to zero. Thus,

$$\tau = \frac{2(1-P^b)P^p}{2(1-P^b)(P^p+(1-\rho)) + P^p(CW_{min}-1)} \tag{3.5}$$

where P^p is the frame generation probability denoting the probability that at least a frame will arrive following a transmission and is given as $P^p = 1-e^{-\lambda T^{cs}}$ and $\rho = \lambda D^T$ is the probability of saturation for stations on the WLAN. D^T is the total delay experienced by the head of line (HOL) frame of ACs and λ is the packet arrival rate.

CW_{min} is the minimum size of the congestion window and this is given as:

$$CW_{min} = 1 + \frac{2(1-P^b)\Big[P^p(1-\tau) - (1-\rho)\tau\Big]}{P^p\tau} \tag{3.6}$$

When the network is in the saturated mode, that is, $P^p = 1$ and $\rho = 1$, τ is given as:

$$\tau = \frac{2(1-P^b)}{2(1-P^b) + CW_{min} - 1} \tag{3.7}$$

and CW_{min} is given as:

$$CW_{min} = 1 + \frac{2(1-P^b)(1-\tau)}{\tau} \tag{3.8}$$

With these probabilities, throughput of a station on the WLAN is defined as

$$S = \frac{P^s m l^{data}}{T^{cs}} \tag{3.9}$$

where

$$T^{cs} = (1-P^b)T^e + nP^sT^s + P^cT^c$$

and l^{data} is the length in bytes of the frames to be transmitted. m is the number of back-to-back frames to be transmitted in the event that fragmentation occurs. In this chapter, we consider network transmitting only on AC type so we set m to one for the AC under consideration since there is not prioritisation of traffic.

T^e is the duration of an empty slot and the value is dependent on the physical layer characteristics of the wireless medium.

T^s is the time for a successful transmission. The request-to-send clear-to-send (RTS/CTS) mode of transmission was used in order to recover quickly and reduce time loss in case of collision and this is given as

$$T^s = T^{RTS} + SIFS + T^{CTS} + DIFS + T^{phy} + T^{mac} + T^{data} + SIFS + T^{ACK}$$

$T^{data} = \frac{l^{data}}{r}$ is the time taken to transmit one MPDU at the PHY data rate r.

T^c is the corresponding duration of collision and this is given as:

$$T^c = T^{RTS} + EIFS$$

3.1.2 Delay Analysis (D^T)

We considered delay as time the head-of-line (HOL) frame spends from the beginning of contending for channel access till the last bit of the packet has been successfully sent. This consists of expected time spent counting down empty slots, expected duration of time when the backoff timer is frozen due to ongoing transmission of another station, expected time spent for retransmission in the event of a collision and time for successful transmission.

Expected Countdown Delay (D^{cd})

This is the time spent during back-off timer countdown i.e. the average number of idle slots a station counts down during the backoff stage without considering times that the backoff counter freezes. For each backoff stage, the average number of slots counted down is given as $T^e \frac{CW_{min}}{2}$ and for countdown process, expected countdown delay is given as:

$$\begin{aligned} D^{cd} =& T^e \Bigg(P^p(1-P^b)(1-\rho)\Big(\sum_{j=1}^{M}(P^{col})^j(1-P^{col})\sum_{h=1}^{j}\frac{CW_{min,h}}{2} + (P^{col})^{M+1}\sum_{h=1}^{M}\frac{CW_{min,h}}{2}\Big) + \\ & \rho\sum_{j=0}^{M}(P^{col})^j(1-P^{col})\sum_{h=0}^{j}\frac{CW_{min,h}}{2} + (P^{col})^{M+1}\sum_{h=0}^{M}\frac{CW_{min,h}}{2}\Bigg) \end{aligned} \tag{3.10}$$

where M is the total number of retransmission and P^{col} is the conditional probability that a transmitted frame of a station experiences collision. This occurs if one of the remaining $(n-1)$ stations attempts transmission and this is

$$P^{col} = 1 - (1-\tau)^{n-1} \tag{3.11}$$

Expected Freezing Delay (D^f)

Freezing delay is the estimated delay experienced by the frame of AC due to freezing its counter when the medium is sensed as busy. This freeze time depends on whether freezing the counter was due to a successful transmission or a collision of another STA on the WLAN. The estimated delay due to a successful transmission is given as:

$$D^{fs} = T^s(n-1)\tau(1-\tau)^{n-2}$$

and the estimated delay due to a collision on the WLAN is:

$$D^{fc} = \Big(1-(1-\tau)^{n-1}-(n-1)\tau(1-\tau)^{n-2}\Big)T^c$$

The expected freezing delay experienced by a station is:

$$D^f = \frac{D^{cd}}{T^e}(D^{fs}+D^{fc}) \tag{3.12}$$

Retransmission Delay (D^{retx})

For a transmitting station, one of the possibly outcome is collision. The delay experienced from collision is estimated as:

$$\begin{aligned} D^{retx} =& T^c\Bigg(P^p(1-P^b)(1-\rho)\Big(\sum_{j=1}^{M} j(P^{col})^j(1-P^{col})+(M+1)(P^c)^{M+1}\Big)+ \\ & \rho\sum_{j=0}^{M} j(P^c)^j(1-P^c)+(M+1)(P^c)^{M+1}\Bigg) \end{aligned} \tag{3.13}$$

Successful Transmission Delay (D^{succ})

The transmission delay is time taken to successful transmit a frame multiplied by the probability that the frame does not collide

$$D^{trans} = T^s(1-(P^{col})^{M+1}) \tag{3.14}$$

In the analysis, we set CW_{min} and CW_{max} as a singular value CW and do not consider number of retransmission in our delay analysis. With this, the average total delay D^T experienced by the HOL frame is then calculated as the sum of the different components analysed above, and with $M = 0$, this gives:

$$D^{cd} = T^e \frac{CW_{min}}{2}$$
$$D^f = \frac{CW}{2}(D^{fs} + D^{fc})$$
$$D^{retx} = T^c P^c$$
$$D^{trans} = T^s(1 - P^c)$$

and

$$D^T = D^{cd} + D^f + D^{retx} + D^{trans} \tag{3.15}$$

3.1.3 WLAN throughput analysis S^{opt}

The control aim is to maintain a pre-defined throughput performance on the WLAN by the AP giving the appropriate value of CW. This is achieved by setting as reference the desired throughput which we term as S^{opt}, for the control system shown in Fig 3.2.

To calculate our reference value, we adopt the method used in [26, 54] which computes the maximum throughput achievable on the WLAN and then gets the corresponding optimal transmission probability τ. We set τ^{opt} as

$$\tau^{opt} = \frac{1}{n}\sqrt{\frac{2T^e}{T^c}} \tag{3.16}$$

and by substituting the optimal transmission probability τ^{opt} in the throughput expression in equation 3.9, we obtain an optimal throughput expression S^{opt} as derived in [11] such that

$$S^{opt} = \frac{ml^{data}}{T^{cs'}} \tag{3.17}$$

where

$$T^{cs'} = T^s + \frac{1-\tau}{K}T^e + T^c\left(\frac{1-\tau}{K}(e^K - 1) - 1\right)$$

and

$$K = \sqrt{\frac{2T^e}{T^c}}$$

3.1.4 Control Algorithm

WLAN transfer function

We represent the WLAN by the notation H(z). In order to represent the WLAN from a control theory point of view, we have to characterise it as a transfer function that takes CW_{min} as input and gives S^{obs} as the output. This representation

should show H(z) as a linear system but from equations (3.8) and (3.17), we see that the relationship between the input and output of the WLAN is not linear. In order to get a transfer function approximation for H(z), we proceed with a linear approximation about the optimal point of operation, noting that the linearised system, $H(z)$, is accurate and stable around the linearised point of operation, therefore:

$$H(z) = \frac{\partial S^{obs}}{\partial CW} = \frac{\partial S^{obs}}{\partial \tau}\frac{\partial \tau}{\partial CW} \tag{3.18}$$

where

$$\frac{\partial S^{obs}}{\partial \tau} = \frac{-\frac{1}{K} m l^{data}\left(T^c(1-e^K) - T^e\right)}{(T^{cs'})^2}$$

$$= -S^{opt}\frac{\frac{1}{K}\left(T^c(1-e^K) - T^e\right)}{T^{cs'}} \tag{3.19}$$

and from equation (3.8)

$$\frac{\partial \tau}{\partial CW} = \frac{P^p\tau^2}{2(1-P^b)\left[\tau\left(b(1-\tau) - P^p - (1-\rho)\right) - \left(P^p(1-\tau) - (1-\rho)\tau\right)\left(\frac{(n-1)\tau}{1-\tau} + \frac{\tau b}{P^p} + 1\right)\right]} \tag{3.20}$$

where

$$b = \lambda n(1-\tau)^{n-1}e^{-\lambda T^{cs}}\left[(1 - \frac{(n-1)\tau}{1-\tau})(T^s - T^c) - (T^e - T^c)\right]$$

By using equations (3.19) and (3.20), we obtain the linearised model for the WLAN transfer function $H(z)$ which is given in equation (3.21).

$$H(z) = \frac{-S^{opt}\frac{1}{K}\left(T^c(1-e^K) - T^e\right)P^p\tau^2}{2(1-P^b)T^{cs'}\left[\tau\left(b(1-\tau) - P^p - (1-\rho)\right) - \left(P^p(1-\tau) - (1-\rho)\tau\right)\left(\frac{(n-1)\tau}{1-\tau} + \frac{\tau b}{P^p} + 1\right)\right]} \tag{3.21}$$

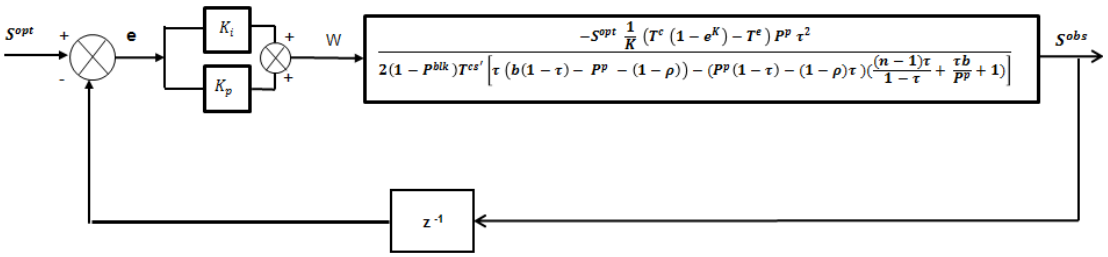

Figure 3.2: Throughput feedback Block Diagram

Controller Design

The controller, C(z), is located in the access point (AP) and it receives as input the control error which is the difference between actual throughput measurement on the WLAN and our design goal and outputs a control signal which is the optimal CW value given the current condition on the WLAN.

From the figure 3.2, we derive the closed loop transfer function G(z), which is given as:

$$G(z) = \frac{C(z)H(z)}{1 + C(z)H(z)z^{-1}}$$

$$= \frac{zH(z)(K_p z - K_p + K_i)}{z(z-1) + H(z)(K_p z - K_p + K_i)}$$

$$= \frac{zH(z)(K_p z - K_p + K_i)}{z^2 + z(H(z)K_p - 1) + H(z)(K_i - K_p)} \tag{3.22}$$

and since our closed loop transfer function is of second order, we equate to the characteristic equation of second order control system, $z^2 + 2\omega_n\xi z + \omega_n^2$, and derive the following expression for the controller variables:

$$K_p = \frac{2\omega_n\xi + 1}{H(z)} \tag{3.23}$$

and

$$K_i = \frac{\omega_n^2}{H(z)} + K_p \tag{3.24}$$

where ω_n is the natural frequency of the system and ξ is the damping ratio, both used as design parameters for the system.

3.1.5 Simulation and performance evaluation

In this section, we evaluate our proposed algorithm through simulations using the control toolbox simulator in Matlab/Simulink Package and Omnet++ simulation tool. We also used simulation parameters shown in Table 3.1. These EDCA parameters are the IEEE 802.11a physical layer parameters. We assume the Request-to-Send Clear-to-Send (RTS/CTS) mode of transmission, we assume $8Mb$ packet arriving at the rate of 1000 packets/seconds. We also design for a controller with 0% overshoot. Since the aim of this work is to maintain pre-defined performance on the network through a PI controller, our simulations revolves around analysis of the effectiveness of our designed controller and the resulting effect on the network. The simulation was carried out with different controller configurations and a step response of the system under varying network conditions was used to show the output of the different controllers. We considered network scenario with 10 nodes that join the network at different time interval so the network is gradually loaded. From equation (3.16), we see that τ varies with the number of nodes transmitting on the network and this enables us to compute the desired throughput used as the reference value S^{opt}. We also derive the representation for the WLAN and with the controller design, we obtained the desired CW value to be used on the WLAN.

Table 3.1: Network parameter values

Parameter	Value	Parameter	Value
T_{SIFS}	$16\mu s$	T^{e}	$9\mu s$
T_{ACK}	$38.67\mu s$	T_{data}	$148\mu s$@$54Mbps$
T_{PHYhdr}	$20\mu s$	$\mathrm{T}_{AIFSmin}$	$34\mu s$
T_{EIFS}	$88.67\mu s$	T_{DIFS}	$34\mu s$
T_{RTS}	46.67 μs	T_{CTS}	38.67 μs
m	1	l^{data}	8000 bits
λ	1000pkt/sec	%OverShoot	0

Table 3.2: Saturation Scenario Results

n	S^{opt}	τ	H(z)	P^c	K_p	K_i	CW
1	22.33	0.3647	-3.0908	0	0.0456	0.0912	8
2	20.38	0.1823	-0.6829	0.1823	0.2065	0.4130	30
3	19.80	0.1216	-0.2868	0.2283	0.4914	0.9834	70
4	19.53	0.0912	-0.1566	0.2493	0.9005	1.8000	120
5	19.37	0.0729	-0.0985	0.2614	1.4316	2.8630	180
6	19.26	0.0608	-0.0675	0.2691	2.0891	4.1780	250
7	19.18	0.0521	-0.0492	0.2746	2.8660	5.7320	350
8	19.13	0.0456	-0.0374	0.2786	3.7700	7.5410	500
9	19.08	0.0405	-0.0294	0.2817	4.7960	9.5930	600
10	19.04	0.0365	-0.0237	0.2842	5.9500	11.8998	800

Table 3.3: Theoretical Unsaturation Scenario at 0.001sec inter-arrival rate

n	H(z) $\rho = 0.1$	H(z) $\rho = 0.3$	H(z) $\rho = 0.5$	H(z) $\rho = 0.7$	H(z) $\rho = 0.9$
1	0.0706	0.0927	0.1349	0.2478	1.5150
2	0.0414	0.0572	0.0923	0.2400	-0.4004
3	0.0329	0.0491	0.0962	2.4276	-0.1045
4	0.0306	0.0517	0.1656	-0.1377	-0.0486
5	0.0323	0.0695	-0.4459	-0.0530	-0.0282
6	0.0393	0.0181	-0.0692	-0.2910	-0.0184
7	0.0626	-0.1368	-0.0327	-0.0186	-0.0130
8	0.3924	-0.0416	-0.0197	-0.0129	-0.0096
9	-0.0676	-0.0224	-0.0134	-0.0096	-0.0074
10	-0.0277	-0.0144	-0.0098	-0.0074	-0.0059

Table 3.2 shows the CW values obtained for a saturated network scenario and Table 3.3 shows the network representation for the different unsaturated network scenarios used in the simulations. Here, it can be seen that the system transfer function was a mixture of negative and positive values. This occurs when the number of stations on the network increases and at the fixed inter-arrival rate, the network gets flooded and saturation sets in. As saturation sets in, the system switches from negative feedback to positive feedback and the controller output actually amplifies the system output in order to achieve the reference set point.

Figure 3.3, shows the sum of throughput obtained for ten contending stations with the CW results obtained compared with the static EDCA protocol values while figure 3.4 shows delay experienced at different packet arrival rate. It can be seen from figure 3.3 that the algorithm was able to maintain a steady throughput value as much as possible while delays experienced by packets are minimal.

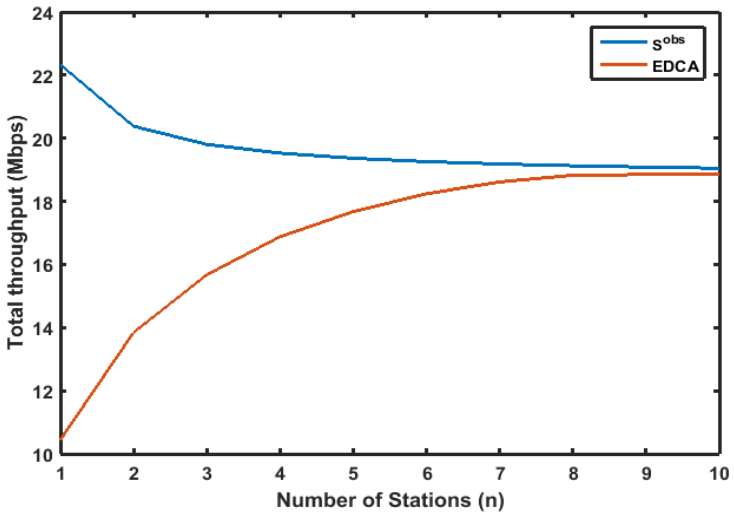

Figure 3.3: Total Throughput

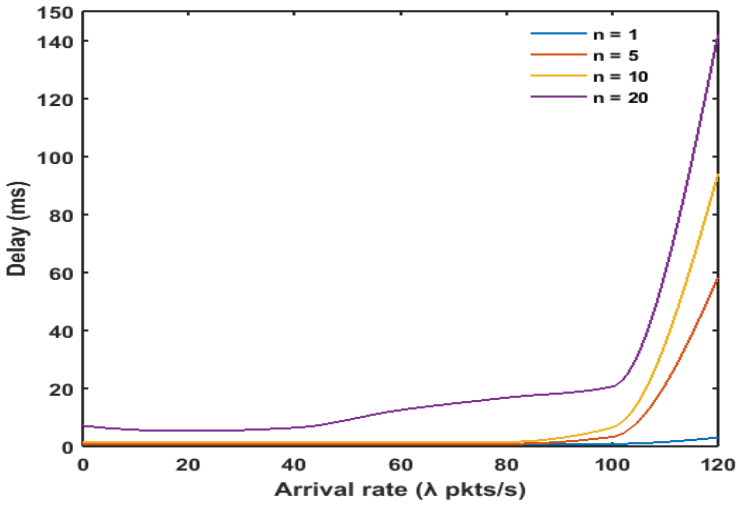

Figure 3.4: Delay

The total air-time used on the WLAN consists of the time for transmissions, both successful transmissions and collisions. This is shown in figure 3.5 and is given as:

$$Air - time = \frac{P^s T^s + \tau P^c T^c}{T^{cs}} \tag{3.25}$$

In figure 3.6, we show how the algorithm responds to change in stations joining and leaving the WLAN. We start with one station at time $t = 0s$, three stations at $t = 30s$, two station at $t = 60s$, four stations at $90s$ and five stations at $t = 120s$. The diagram shows that the algorithm was stable and adapt to changes.

3.1.6 Stability Analysis and Controller Performance

In this section, we analyse the stability and performance of the controller. For stability, we use the Routh-Hurwitz stability criterion and for performance, we access the settling time and the associated steady state error.

In figure 3.7 and figure 3.8, we show system output and error signal diagram for the case of 2 nodes on the WLAN. These figures shows that the steady state was reached in about 10 seconds and the controller was able to drive the WLAN to operate with a steady state error of 0. The controller maintained steady state

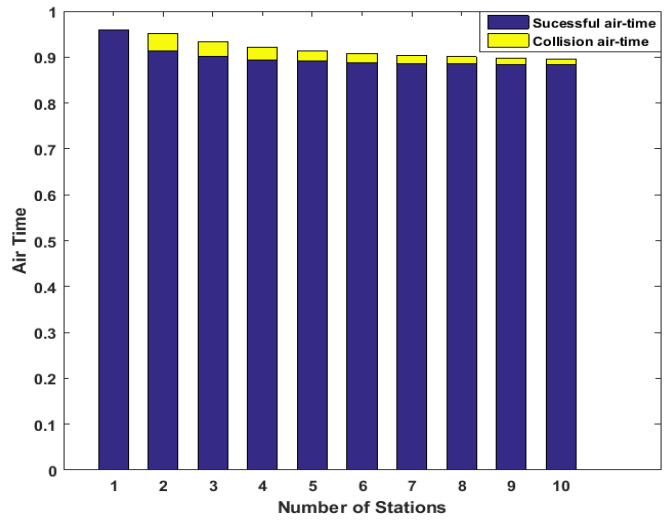

Figure 3.5: Station Air-time

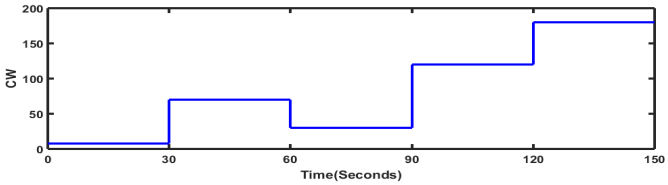

Figure 3.6: Contention window over time

without oscillations as required by specifying 0% overshoot and the system was able to operate at steady state condition. This confirms the accuracy of the equivalent block transfer function since steady state error is zero when $G(1) = 1$. Due to the nature of the Wlan to function in unsaturated and saturated mode, we noticed that the controller output at saturation can be a negative value integer so the access point will in such a case output the absolute integer value.

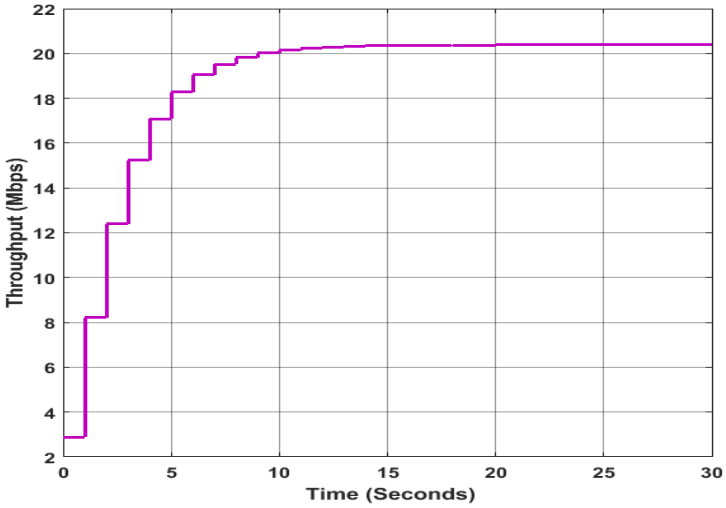

Figure 3.7: System output for n=2

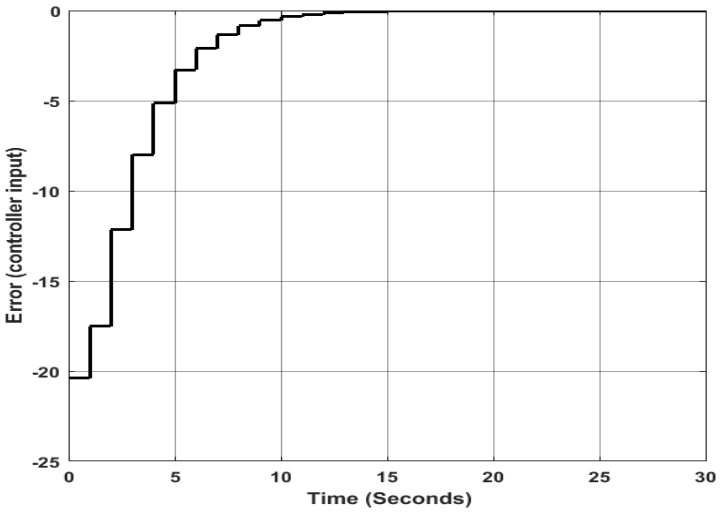

Figure 3.8: Error Signal for n=2

From the closed loop transfer function, equation (3.22), it can be seen that there are two roots to the denominator, thus two zero locations - $z_{0,1}$ which is fixed at the origin and $z_{0,2} = 1 - \frac{K_i}{K_p}$. Figures 3.9 to 3.16 shows different locations of $z_{0,2}$ and K_p and K_i are tuned and its effect on transient response and gain margin of the system. When z_2 is located on the positive real axis, as shown in figure 3.9, we have a controller situation where $K_p > K_i$. The resulting poles are real with one on the positive axis and the other on the negative axis. The gain margin, fig 3.10, increases with greater distance between the pole and zero location giving more

room for gain adjustments/tuning. Optimal configuration is achieved at location $z_2 = 0.067$ which corresponds to a system that is critically damped for a 20ms settling time. In this situation, the controller will not experience any oscillations and its mode of operation is equivalent to a first order control system.

When location of the zero moves to the negative real axis, a second order system results where the closed loop pole locations can either be on the real axis or they could be complex conjugates and controller situation is $K_p < K_i$. Figure 3.11, figure 3.12 and figure 3.13 shows the system output, root locus and Bode plot when the poles are strictly on the real axis. Maximum gain was achieved at $z_2 = -0.33$. Further left from the -0.33 mark, the gain margin begins to drop.

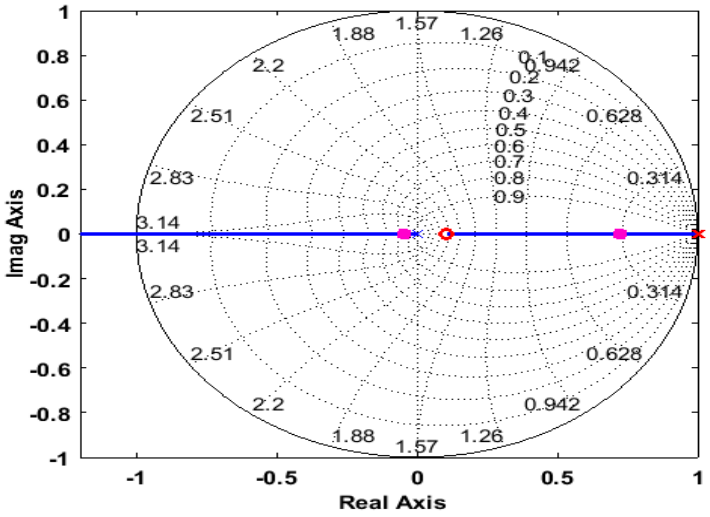

Figure 3.9: Root locus for $K_p > K_i$

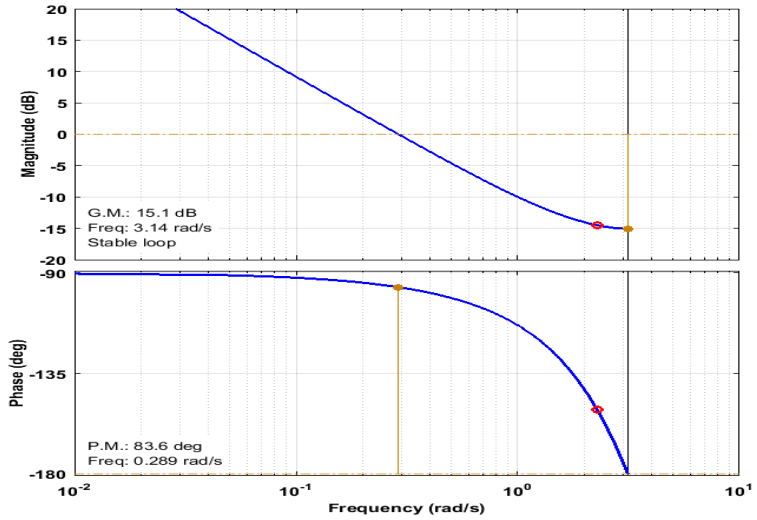

Figure 3.10: Bode plot for $K_p > K_i$

When the close loop poles leave the real axis and move into the complex axis region, the system breaks into oscillation before settling as shown in fig 3.14 where we show the system output for a pair of complex poles. We maintained $z_2 = -0.33$ and moved the poles to the complex plane. Here, transient response is faster but with overshoots and as the pole location moves closer to the z_2 location, we had lowest records of gain margin. Figures 3.15 and 3.16 show the corresponding root locus and bode plot diagrams respectively.

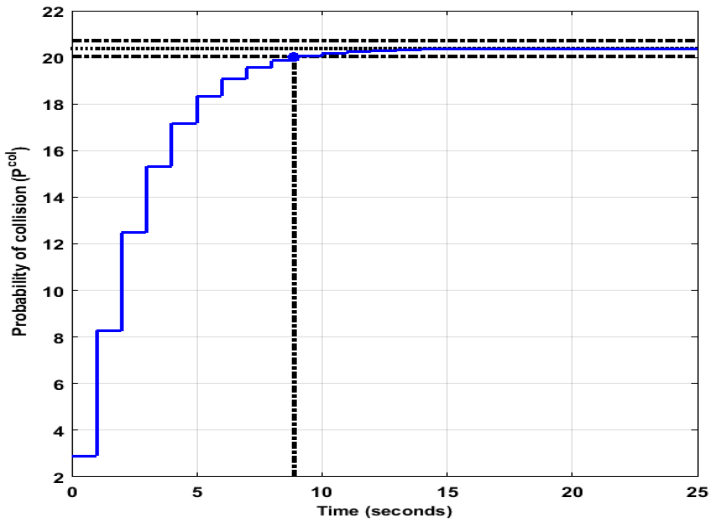

Figure 3.11: Step response for $K_p < K_i$ with real poles

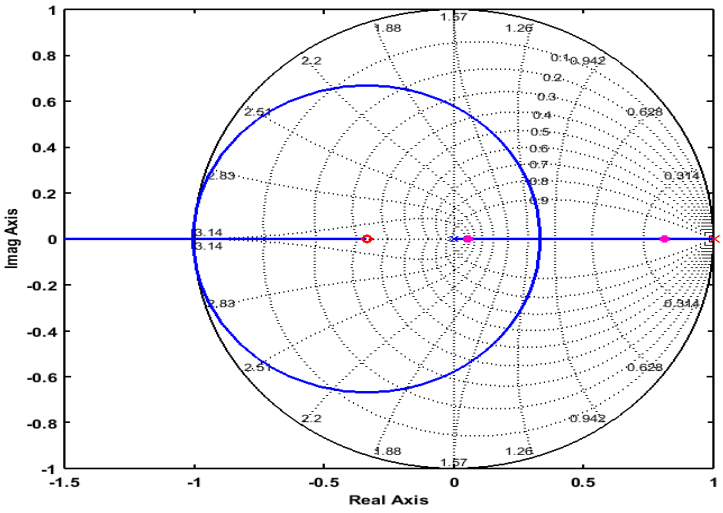

Figure 3.12: Root locus plot for $K_p < K_i$ with real poles

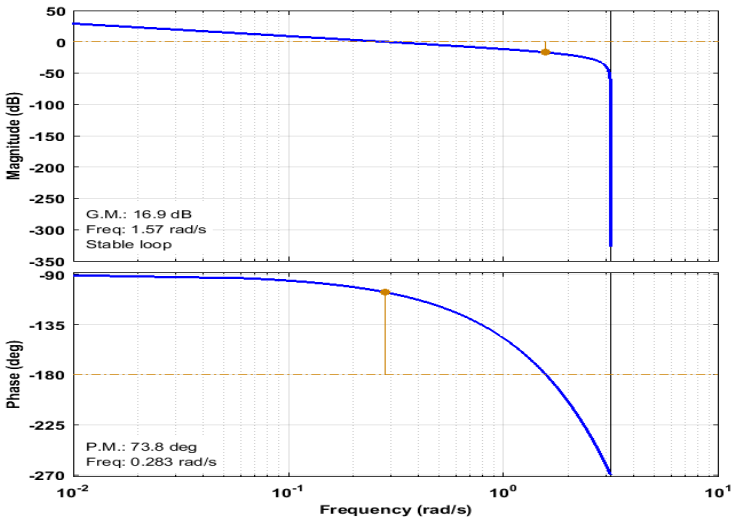

Figure 3.13: Bode plot for $K_p < K_i$ with real poles

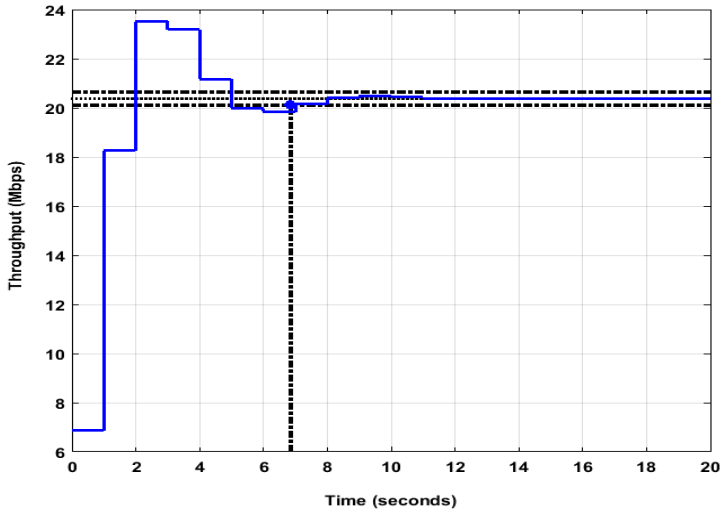

Figure 3.14: Step response for $K_p < K_i$ with complex poles

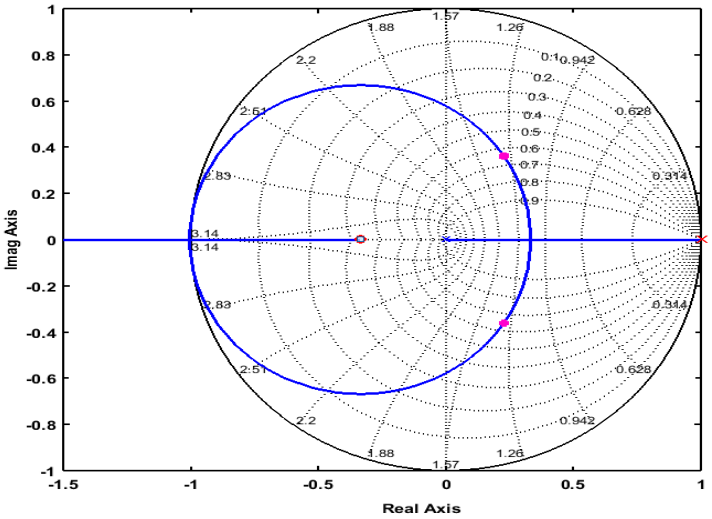

Figure 3.15: Root locus plot for $K_p < K_i$ with complex poles

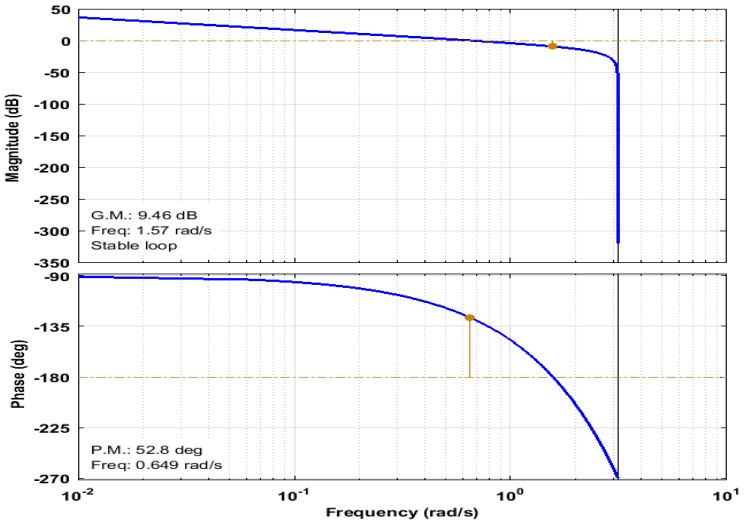

Figure 3.16: Bode plot for $K_p < K_i$ with complex poles

3.2 Probability of Collision Control

In section 3.1, one of the observations made during simulation was the relationship between the WLAN system and the reference set point. It was observed that with high set point values and low WLAN transfer functions, the controller output was

a high figure that could get out of the range of the EDCA values stipulated in the IEEE standards. With this, we commence this section by controlling probability of collision, which is a value that is of closer range to the system transfer function and we analyse the behaviour of the system and controller from this point of view.

3.2.1 Optimal collision probability P^{opt}

In this section, we use the optimal collision probability, P^{opt}, derived in [54] as shown below. We substitute equation 3.16 into equation 3.11 which yields:

$$P^{opt} = 1 - \left(1 - \frac{1}{n}\sqrt{\frac{2T^e}{T^c}}\right)^{n-1}$$

This equation can then be approximated as

$$P^{opt} \approx 1 - e^{-\sqrt{\frac{2T^e}{T^c}}} \tag{3.26}$$

3.2.2 WLAN transfer function

The WLAN transfer function is represented as H(z). H(z) is the linear relationship between the input CW and output P^{col} of the WLAN. From equations 3.8 and 3.11, we see that the relationship between the input and output of the WLAN is not linear so we proceed first by writing the non-linear differential equation representing the two variables and then linearising about a stable point of operation, which in this case is the optimal point P^{opt}, in order to obtain a suitable transfer function.

To achieve this, we proceed as follows:

Let

$$CW - CW_{opt} = \partial CW$$

and

$$P^{col} - P^{opt} = \partial P$$

where ∂ is for small displacements about the optimal point. If H(z) represents the slope of the line, then

$$H(z) = \frac{\partial P^c}{\partial CW} = \frac{\partial P^c}{\partial \tau}\frac{\partial \tau}{\partial CW}$$

where

$$\frac{\partial P^c}{\partial \tau} = (n-1)(1-\tau)^{n-2}$$

By using these two derivatives, we obtain the linearised model for the WLAN

transfer function $H(z)$.

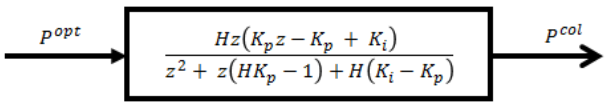

Figure 3.17: Equivalent Block Diagram

3.2.3 Simulation and performance evaluation

In this section, we evaluate our proposed algorithm while controlling the probability of collision on the network and we used simulation parameters shown in Table 3.1. Since the aim of this work revolves around the design of a controller, our simulations show the analysis of the effectiveness of our designed controller.

Table 3.4: Saturation Scenario Results

n	P^{opt}	τ	H(z)	K_p	K_i	CW
1	0.0	0.3647	0	0	0	0
2	0.1823	0.1823	-0.0217	6.4983	12.9956	14
3	0.2283	0.1216	-0.0158	8.9249	17.8497	19
4	0.2493	0.0912	-0.0121	11.654	23.3079	25
5	0.2614	0.0729	-0.0097	14.537	29.0748	31
6	0.2691	0.0608	-0.0081	17.4090	34.8180	37
7	0.2746	0.0521	-0.0070	20.1447	40.2894	43
8	0.2786	0.0456	-0.0061	23.1169	46.2338	50
9	0.2817	0.0405	-0.0054	26.1135	52.2270	55
10	0.2842	0.0365	-0.0049	28.7782	57.5563	62

Table 3.5: Theoretical Unsaturation Scenario at 0.001sec inter-arrival rate

n	H(z) $\rho = 0.1$	H(z) $\rho = 0.3$	H(z) $\rho = 0.5$	H(z) $\rho = 0.7$	H(z) $\rho = 0.9$
1	0	0	0	0	1
2	0.0051	0.0070	0.0113	0.0294	-0.0491
3	0.0074	0.0110	0.0216	0.5445	-0.0234
4	0.0099	0.0167	0.0534	-0.0444	-0.0157
5	0.0135	0.0292	-0.1872	-0.0223	-0.0118
6	0.0203	0.0939	-0.0357	-0.0150	-0.0095
7	0.0384	-0.0838	-0.0200	-0.0114	-0.0079
8	0.0782	-0.0295	-0.0140	-0.0092	-0.0068
9	-0.0544	-0.0180	-0.0108	-0.0077	-0.0060
10	-0.0250	-0.0130	-0.0088	-0.0066	-0.0053

We carried out simulations with different configurations of the controller and a step response of the system under varying network conditions was used to show the output of the different controller configurations.

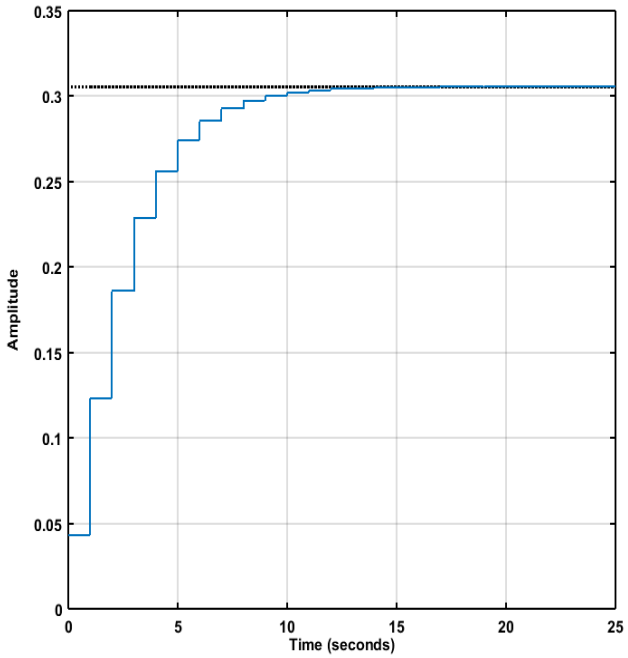

Figure 3.18: Step response for $K_p > K_i$

In figures 3.18 to 3.20, we show system output, error signal and root locus diagram for the case of 2 nodes on the WLAN. The controller was able to drive the WLAN to operate with a steady state error of 0 in about 10ms and the system maintained optimal collision probability on the WLAN. This confirms the accuracy of the equivalent block transfer function since steady state error is zero iff:

$$G(1) = 1$$

3.2.4 Controller Performance and Stability Analysis

We also considered two locations of the zero,z_2, of the closed loop transfer function and its effect on transient response and gain margin of the system. When z_2 is located on the positive real axis, we have a controller situation where $K_p > K_i$. The resulting poles are real with one on the positive axis and the other on the negative axis. The gain margin increases with greater distance between the pole and zero location giving more room for gain adjustments/tuning. Optimal configuration is a trade off between settling time, gain and phase margin values. Figures 3.19 and 3.20 shows the root locus and Bode plots for this scenario.

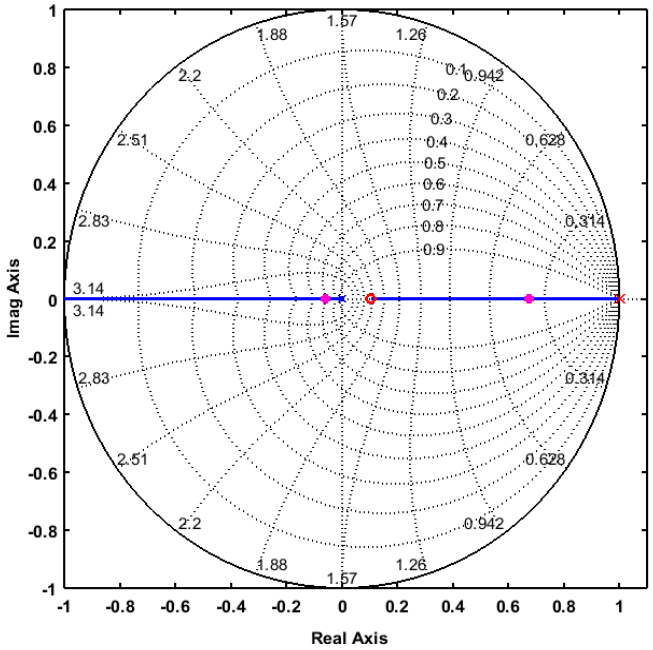

Figure 3.19: Root locus for $K_p > K_i$

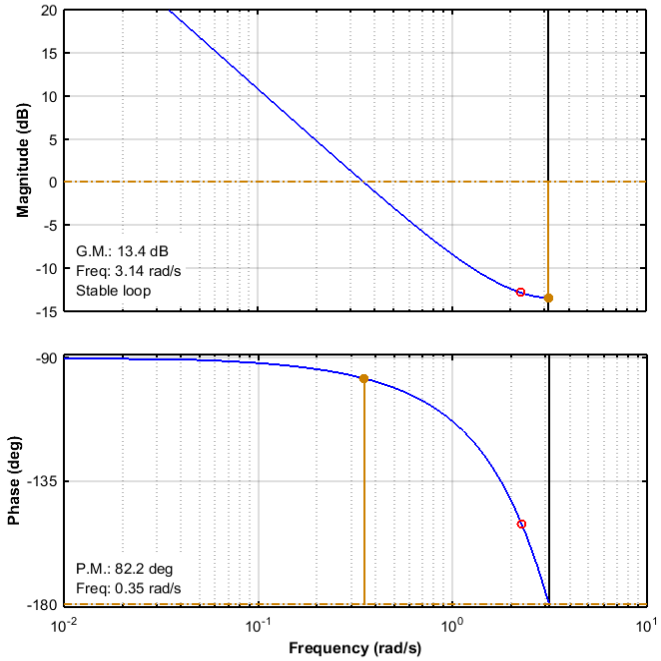

Figure 3.20: Bode Plot for $K_p > K_i$

As discussed in section 3.1.6, when the location of the zero moves to the negative real axis, a second order system results where the closed loop pole can either be on the real axis or they could be complex conjugates and controller situation is $K_p < K_i$. Figures 3.21, 3.22, 3.23 shows the system output, root locus and Bode plot when the poles are strictly on the real axis and $z_2 = -0.33$.

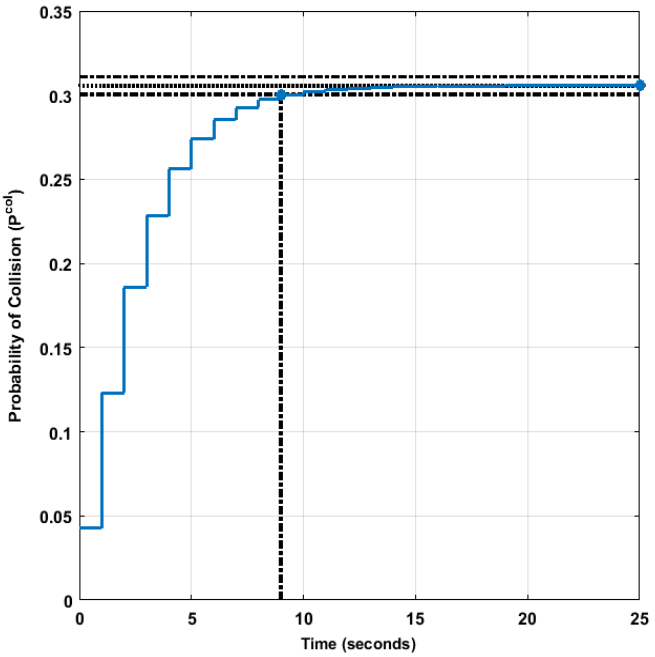

Figure 3.21: Step response for $K_p < K_i$ with real poles

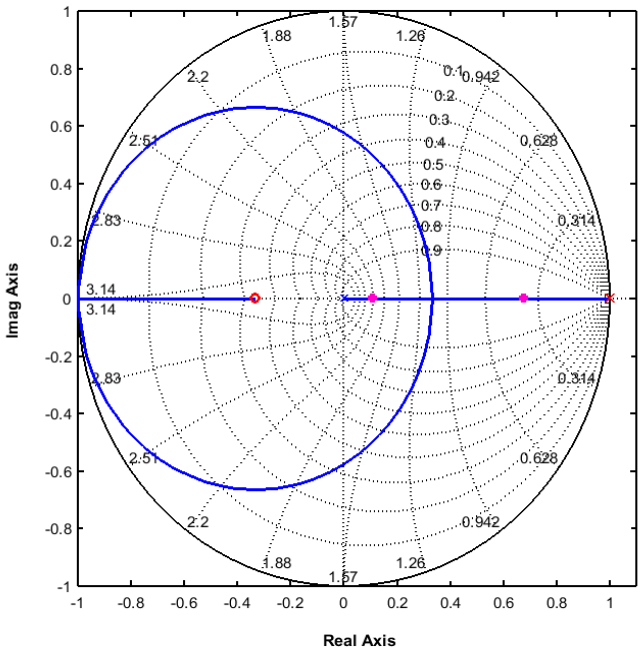

Figure 3.22: Root locus for $K_p < K_i$ with real poles

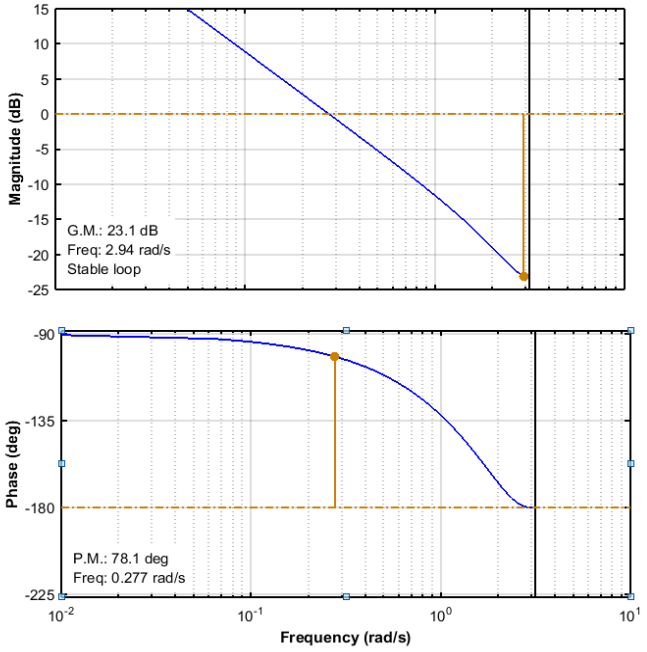

Figure 3.23: Bode Plot for $K_p < K_i$ with real poles

In fig 3.24, we show the system output for a pair of complex poles. We maintained $z_2 = -0.33$ and moved the poles to the complex plane. Here, transient response is faster but with overshoots and as the pole location moves closer to z_2, we had lowest records of gain margin. Figures 3.25 and 3.26 shows the corresponding root locus and bode plot diagrams respectively.

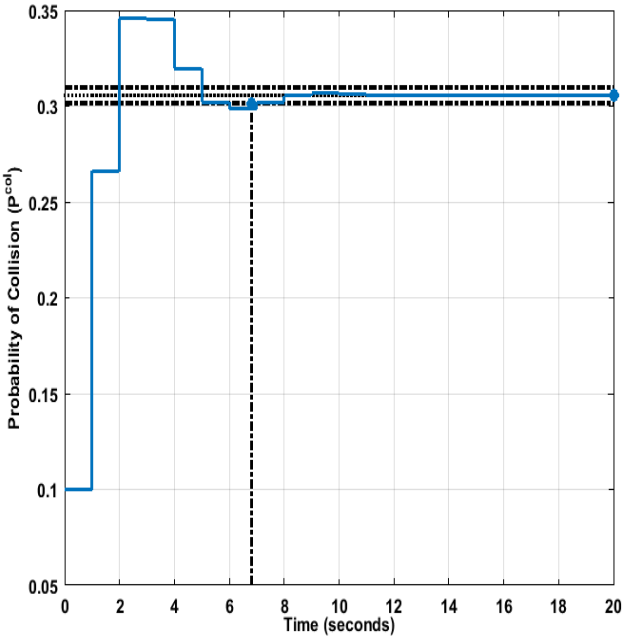

Figure 3.24: System output for $K_p < K_i$ with complex poles

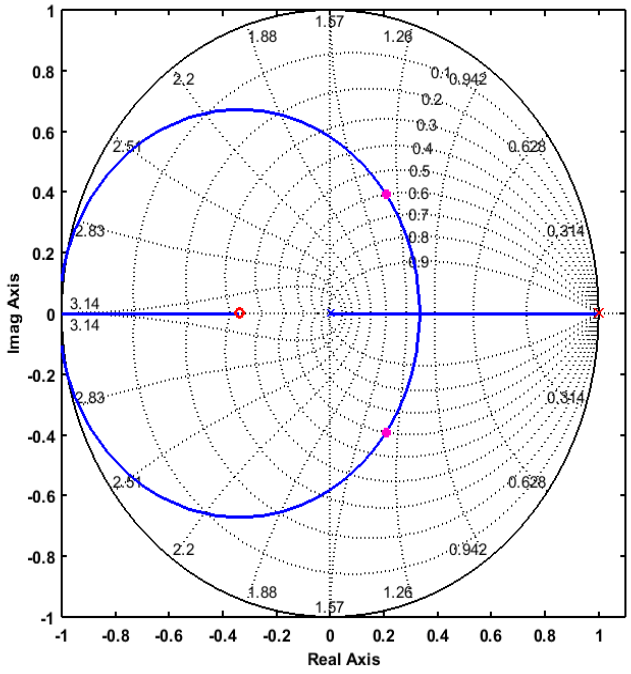

Figure 3.25: Root locus for $K_p < K_i$ with complex poles

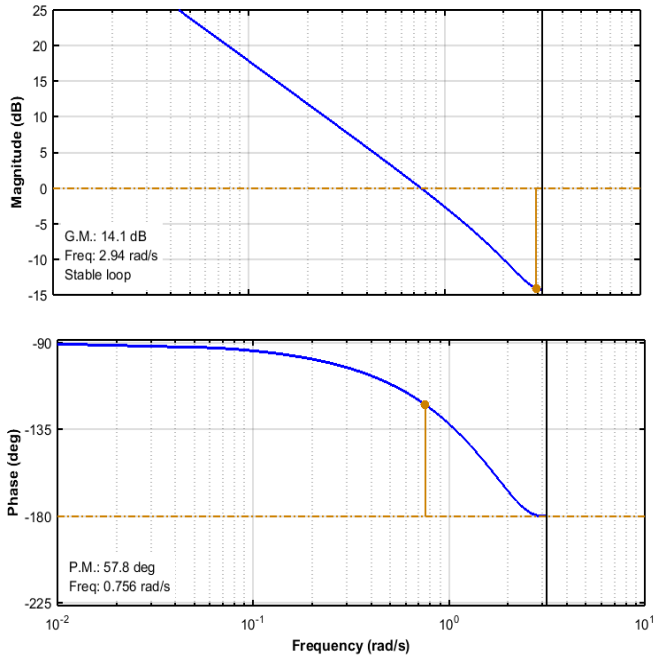

Figure 3.26: Bode plot for $K_p < K_i$ with complex poles

3.3 Discussion and Conclusion

The principle of operation for the algorithm works based on the fact that based on network scenario, an appropriate contention window value should be given by the access point in its beacon frame for stations to use during the contention phase of the channel access process. Without necessarily knowing the actual number of stations on the network, the access point monitors level of activity on the network and depending on the network parameter being controlled, the AP can give out the right CW value to be used.

From the results in sections 3.1.6 and 3.2.4, it can be seen that the probability of collision is the preferred variable to be controlled as it is possible to have a fixed reference value irrespective of network scenario and the CW outputs are closer in value to the recommended values in the IEEE 802.11e standards, compared to using throughput as the control variable.

We have derived the optimal collision rate on a network based on number of contending stations in section 3.2.1. To measure the actual collision probability on the WLAN - the observed collision probability (P^{obs}), we use the same principle in [1], [19], [53] and [54], where a station measures the total number of RTS frames sent within the beacon interval of $100ms$. Assuming that a packet collides with a constant collision probability on the WLAN, estimated number of missing CTS frames - $E(N^{M-CTS})$ shows rate of collision on the WLAN. This is computed by the transmitting station by taking record of the total number of missing CTS

frames and this is sent to the AP after every $100ms$, thus

$$E(N^{M-CTS}) = N^{RTS}P^{col}$$

where N^{RTS} is number of RTS frames sent out by a station. The observed probability of collision is then estimated as

$$E\left(\frac{N^{M-CTS}}{N^{RTS}}\right) = P^{obs}$$

In conclusion, we have looked at the design of a PI controller for use in tuning the contention window parameter on the WLAN while controlling the collision rate on the WLAN. We have also optimized the controller parameters. Particularly, we focused on the different possible transient response of the controller and the importance of proper tuning of the controller in order to achieve design requirements and optimal gain margins. In the following chapters, we will apply the control algorithm to network scenarios with multiple ACs on the network for the purpose of improving throughput while maintaining fairness on the WLAN.

Chapter 4

Multiple AC Network with Decentralised Control

On the IEEE 802.11e network, it is very common to have network scenarios where more than on traffic type traverses the network. Therefore, in this chapter, we focus on the application of the feedback algorithm on an ideal WLAN with multiple network traffic on the network. Considering that the input to the network or system is the minimum contention window value and the output is the probability of collision, the WLAN is then classified as a multiple-input-multiple-output ($MIMO$) system because each AC has a specific input-output pair attached to it. For a MIMO network, interference occurs due to the dependencies in the probability modelling structure and the effect of this is shown when the controller output of one AC category influences the performance of another AC making controller tuning difficult. Our contribution in this chapter is in the use of the decoupling control technique to counteract these influences so that it is then possible to control each ACs input and output pair independently without interferences.

To accomplish this, modelled the network with a fairness criteria that is constrained to the ratio of throughput among the different traffic types and delay to be maintained among delay sensitive traffic. We then designed the network controller which for each traffic type and a decoupling controller that counters the interferences allowing each traffic type to be controlled independently.

Table 4.1: EDCA contention parameter values

AC_i	AIFSN	CW_{min}	CW_{max}	$TXOP_{limit}$
AC_{voice}	2	7	15	0.003008
AC_{video}	2	15	31	0.006016
$AC_{besteffort}$	3	31	1023	0
$AC_{background}$	7	31	1023	0

4.1 WLAN model

In order to define the wireless network, we consider a WLAN with an access point (AP) and a number of stations (STAs). We consider N different ACs and assume each station transmits a single traffic type. We also assume n_i stations for the ith AC and the total number of stations on the network is $n = \sum_{i=0}^{N-1} n_i$. We used the RTS/CTS mode of transmission in order to reduce time loss during collision and we also assume an error free channel. We define the probabilities associated with the contention slot as follows:

$$P_i^s = \tau_i(1-\tau_i)^{n-1} \prod_{\substack{j=0 \\ j\neq i}}^{N-1} (1-\tau_j)^{n_j}$$

where P_i^s is the probability of successful transmission of a frame of AC_i.
The total probability of a successful transmission on the network, P^s, is given as:

$$P^s = n_i \sum_{i=0}^{N-1} P_i^s = P^e \sum_{i=0}^{N-1} \frac{n_i \tau_i}{1-\tau_i}$$

where P^e is the probability that the channel is idle, and is given as

$$P^e = \prod_{i=0}^{N-1} (1-\tau_i)^{n_i}$$

Collision probability within a slot time is the probability that the slot is busy but not with a successful transmission and this is represented as

$$P^c = 1 - P^e - P^s$$

τ_i is the probability that AC_i transmits in a randomly chosen slot. We have calculated this parameter from equation 5.3 in chapter 3. Thus

$$\tau_i = \frac{2(1-P_i^{blk})P_i^p}{2(1-P_i^{blk})(P_i^p+(1-\rho_i))+P_i^p(CW_{min,i}-1)} \tag{4.1}$$

where P_i^{blk} is the blocking probability [36] associated with the inability of a CAF to resume count down immediately after a busy channel due to it $AIFS_i$ value. $AIFS_i = t_i \times T^e + SIFSTime$, where t_i is the number of empty slot time AC_i has to wait and $SIFS$ is the duration of the Short Inter-Frame Space. P_i^{blk} is given as

$$P_i^{blk} = 1 - \left[(1-\tau_i)^{n_i-1}\prod_{\substack{j=0\\ j\neq i}}^{N-1}(1-\tau_j)^{n_j}\right]^{t_i-t_{min}+1} \tag{4.2}$$

where t_{min} is the time associated with the smallest $AIFS_i$ value. $CW_{min,i}$ is the minimum size of the congestion window for AC_i, P_i^p is the frame generation probability [36] and is given as $P_i^p = 1 - e^{-\lambda_i T^{cs}}$ and $\rho_i = \lambda_i D_i^T$ is the probability of saturation for AC_i and D_i^T is the delay experienced by the head-of-the-line packet.

4.1.1 Throughput analysis

The throughput for a given AC_i on the WLAN is the rate at which its frame is successfully transmitted on a WLAN. With the assumption of equal size frame, L, for each transmission and same physical layer transmission rates on the network, throughput is given as

$$S_i = \frac{P_i^s m_i L}{P^e T^e + \sum_{i=0}^{N-1} n_i P_i^s T_i^s + (1-P^s-P^e)T^c} \tag{4.3}$$

where

m_i is the number of packets of AC_i transmitted in one TXOP

T^e, T_i^s, T^c are time intervals for an empty slot, a successful transmission and collision respectively and they are given as:

$$T_i^s = T^{rts} + SIFS + T^{cts} + AIFS_i + m_i(T^{phy} + SIFS + \frac{L}{C} + SIFS + T^{ACK})$$

$$T^c = T^{rts} + EIFS$$

4.1.2 Delay Analysis

As in section 3.1.2, we considered delay as time the Head of Line (HOL) frame spends from the beginning of contending for channel access till the last bit of the packet has been successfully sent and only successful transmissions are considered. We analyse using the same approach and in addition, we consider multiple traffic on the network which changes the network dynamics. Since our approach is also for the access point (AP) to broadcast the optimal CW value, we set $CW_i^{min} = CW_i^{max}$=CW_i and do not consider number of retransmission in our delay analysis.

- *Expected Countdown Delay (D_i^{cd})*: This stipulates time spent in back-off timer countdown i.e. the average number of idle slots a station counts down during backoff stage without considering times that the backoff counter freezes. For any backoff stage, the average number of slots counted down is given as

$$D_i^{cd} = T^e \frac{CW_i}{2} \tag{4.4}$$

- *Expected Blocking Delay (D_i^{blk})*: Blocking delay is the estimated delay experienced by a frame due to freezing its counter when the medium is sensed as busy. This freeze time varies with each AC_i due to their different $AIFSN_i$. A station is blocked due to either a successful transmission or a collision on the WLAN and the associated delay is

$$D_i^{blk} = D_i^{cd} P_i^{blk} \frac{P^s T^s + (P^b - P^s)T^c}{P^b T^e} \tag{4.5}$$

  where P^b is the probability of a busy channel which is given as

$$P^b = 1 - \prod_{i=0}^{N-1} (1 - \tau_i)^n$$

- *Collision Delay (D_i^{col})*: For a STA awaiting transmission, there is some delay experienced if collision of the RTS/CTS frames occur and we estimate this

time as:

$$D_i^{col} = T^{col} P_i^{col} \tag{4.6}$$

where P_i^{col} is the conditional probability that a frame of AC_i experiences collision. This is represented as

$$P_i^{col} = 1 - (1 - \tau_i)^{n_i - 1} \prod_{j=0, j \neq i}^{N-1} (1 - \tau_j)^{n_j} \tag{4.7}$$

- *Transmission Delay (D_i^{trans})*: The transmission delay is time taken to successful transmit a frame multiplied by the probability that the frame does not collide

$$D_i^{trans} = T_i^s (1 - P_i^{col}) \tag{4.8}$$

The average total delay D_i^T experienced by the HOL frame is then calculated as the sum of the different components analysed above, which gives:

$$D_i^T = D_i^{cd} + D_i^{blk} + D_i^{col} + D_i^{trans}$$

$$\begin{aligned} D_i^T = & \frac{CW_i}{2}(T^e + T^{col}) + \alpha(T_i^s - T^c)(\frac{CW_i(n-1)}{2(1-\tau_i)} + 1) + \frac{CW_i}{2}(\alpha n \sum_{\substack{j=0 \\ j \neq i}}^{N-1} \frac{\tau_j}{1-\tau_j} \\ & \sum_{\substack{j=0 \\ j}}^{N-1} (T_j^s - T^{col})) - T^{col}(1-\tau_i)^{n-1} \prod (1-\tau_j)^n + T^{col} \end{aligned} \tag{4.9}$$

where

$$\alpha = (1 - \tau_i)^{n_i - 1} \prod_{\substack{j=0 \\ j \neq i}} (1 - \tau_j)_j^n$$

4.1.3 Fairness Criteria

In order to establish fairness in throughput allocation among the different traffic types on the network, the ratio of throughput obtained on the network is used as weights that ensure the lowest priority traffic gets to transmit its packets and is not disadvantaged among other higher priority flows. Also, some delay constraints were applied to ensure that delay sensitive traffic are able to meet the

QoS requirements. The following optimization task is defined where the optimal transmission probability is obtained subject to the set constraints. This optimal value is then used to set the operating collision probability on the network. The optimization task is defined as follows:

$$\begin{aligned} \text{Maximize:} \quad & U = n \sum_{i=0}^{N-1} S_i(\tau_i) \\ \text{Subject to:} \quad & D_i^T(\tau) \leq m_i d_i \qquad 0 \leq i \leq N-1 \\ & \frac{S_i}{S_k} = c_i, ..., \frac{S_{N-1}}{S_k} = c_{N-1} \qquad 0 \leq i \leq N-1 \end{aligned}$$

where
$m_i d_i$ is the desired delay constraint for the TXOP burst of AC_i,
S_k is the throughput of the lowest priority traffic on the network

4.2 Feedback control Algorithm

In this section, we design our algorithm using the concepts of feedback control. The aim is to design a control law that feedback the collision probability measured on the WLAN, compares it with the reference collision probability and then outputs a CW value that achieves the QoS design goal. This is represented by fig 4.3. We obtain the set reference value by calculating the optimal collision rate using the optimal transmission probability τ_{opt} obtained above.

4.2.1 WLAN transfer function

The WLAN transfer function is obtained through a linear relationship between the input CW_i and output P_i^{col} of the WLAN.

From equations (4.1) and (4.7), we see that the relationship between the input and output of the WLAN is non-linear and each output P_i^{col} is a function of the transmission probability of the different ACs on the network. So, we proceed first by writing the non-linear differential equation representing this relationship and then linearize about a stable point to obtain a suitable transfer function.

This is obtained as follows:

Let

$$CW_i - CW_i^{opt} = \partial CW_i$$

and

$$P_i^{col} - P_i^{opt} = \partial P_i^{col}$$

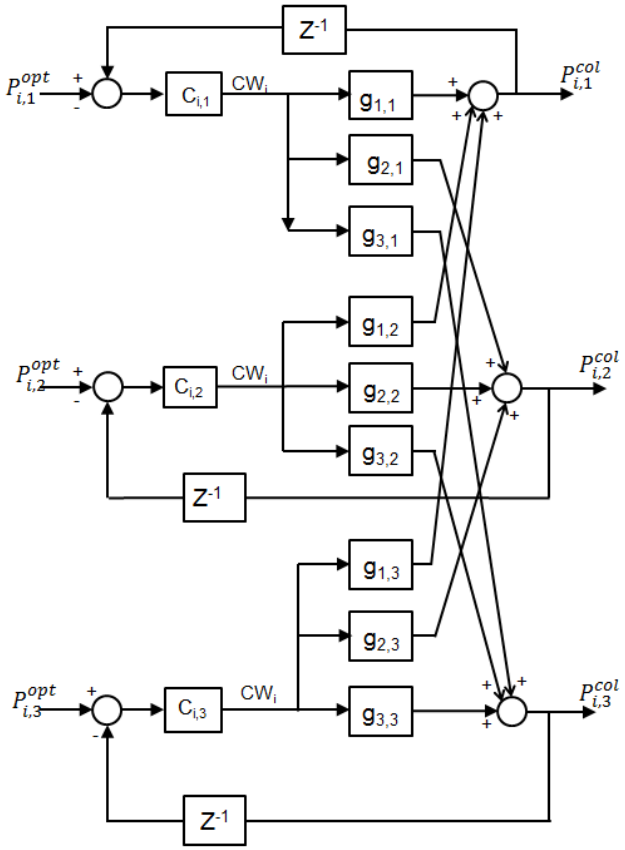

Figure 4.1: Coupled layout

where ∂ is for small displacements about the steady state point, and if H(z) represents the slope of the curve at P^{opt} , then

Solving for the slope (z) , we have

$$\frac{\partial P_i^{col}}{\partial CW_i} = \sum_{k=0}^{N-1} \frac{\partial P_i^{col}}{\partial \tau_k} \frac{\partial \tau_k}{\partial CW_i}$$

$$\frac{\partial P_i^{col}}{\partial CW_j} = \sum_{k=0}^{N-1} \frac{\partial P_i^{col}}{\partial \tau_k} \frac{\partial \tau_k}{\partial CW_j} \tag{4.10}$$

where

$$\frac{\partial P_i^{col}}{\partial \tau_i} = \frac{n_i - 1}{(1-\tau_i)^2} \prod_{h=0}^{N-1} (1-\tau_h)^{n_h}$$

$$\frac{\partial P_i^{col}}{\partial \tau_j} = \frac{n_i - 1}{(1-\tau_i)(1-\tau_j)^2} \prod_{h=0}^{N-1} (1-\tau_h)_j^n$$

and

$$\frac{\partial \tau_i}{\partial CW_i} = \frac{\tau_i^2}{-2(1-P_i^{blk})\left[(n_i-1)(A_i - A_{min}+1)\tau_i + 1\right]}$$

$$\frac{\partial \tau_i}{\partial CW_j} = \frac{\tau_i(1-\tau_j)P_i^P}{2n_j(1-P_j^{blk})(A_j - A_{min}+1)(P_j^P(1-\tau_j)+(1-\rho_j)\tau_j)}$$

From equation 5.5, we obtain a linearised but coupled transfer function matrix of the WLAN.

$$G(z) = \begin{pmatrix} \frac{\partial P_0^{col}}{\partial CW_0} & \frac{\partial P_0^{col}}{\partial CW_1} & \cdots & \frac{\partial P_0^{col}}{\partial CW_{N-1}} \\ \frac{\partial P_1^{col}}{\partial CW_0} & \frac{\partial P_1^{col}}{\partial CW_1} & \cdots & \frac{\partial P_1^{col}}{\partial CW_{N-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_{N-1}^{col}}{\partial CW_0} & \frac{\partial P_{N-1}^{col}}{\partial CW_1} & \cdots & \frac{\partial P_{N-1}^{col}}{\partial CW_{N-1}} \end{pmatrix}$$

4.2.2 Decoupling Control

The system $G(z)$ is a multi-variable (MIMO) system where the controller output for an AC influences the modelled systems of other ACs. The interference needs to be decoupled in order to treat each AC separately and give a specific CW output to each traffic type. To achieve this, we use the decoupling concept in [5], [24] where the multi-variable system $G(z)$ is reduced to a diagonal matrix for better control with individual input-output pairing. The resulting system, $H(z)$, is a single-input single-output (SISO) system achieved using a dynamic decoupler signal $D(z)$ shown in fig 4.2. In designing the decoupling controller, the goal is to achieve an equivalent WLAN transfer function $H(z)$ that is a diagonal matrix such that:

$$H(z) = G(z) \cdot D(z) \tag{4.11}$$

where:

$$D_{i,j} = 1, \forall i = j$$

and the re-shaped system transfer function $H(z)$ is then given as:

$$H(z)_{N\times N,diag} = G(z)_{N\times N} D(z)_{N\times N} \tag{4.12}$$

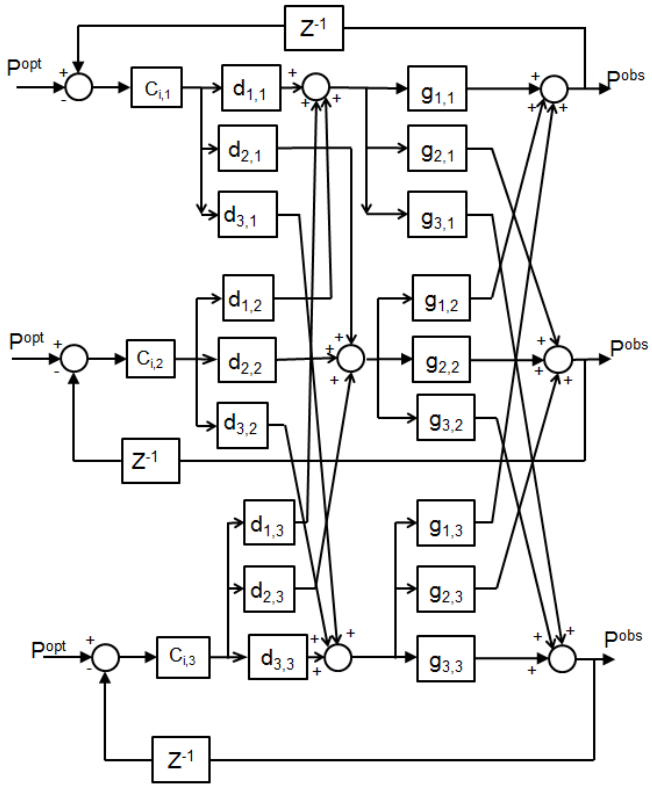

Figure 4.2: Schematic of a 3x3 system

If $H(z)$ is diagonal, the following expression holds

$$D(z) = G(z)^{-1}.H(z)$$

and omitting the discreet notation z, we have

$$D(z) = \frac{1}{|G|}\begin{pmatrix} g_{11}h_1 & g_{12}h_2 & \cdots & g_{1n}h_n \\ g_{21}h_1 & g_{22}h_2 & \cdots & g_{2n}h_n \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1}h_1 & g_{n2}h_2 & \cdots & g_{nn}h_n \end{pmatrix} \tag{4.13}$$

where g_{11} is the adjoint of the G(z) matrix.
Our design task is solvable if the elements d_{ii} of the D(z) matrix equates to one so that the control signal specific to each AC is maintained and the coupling effect of the other ACs is cancelled out. This implies that the value of the diagonal matrix, H(z), can be obtained as

$$h_{ii} = \frac{|G|}{g_{ii}} \tag{4.14}$$

Having obtained the decoupling matrix, the control task can now be treated as

a SISO system.

4.2.3 PI Controller configuration

The next task is to obtain the controller configuration. For the feedback control loop to be stable and without steady state errors, the designed controller must include an integrator. The controller matrix is of the form:

$$C(z) = \begin{pmatrix} c_{i,i} & 0 & \cdots & 0 \\ 0 & c_{i,i} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_{i,i} \end{pmatrix}$$

where $c_{i,i}$ has a general expression given as:

$$c_{i,i} = K_p + \frac{K_i}{z-1}$$

From fig 4.3, we obtain the closed loop transfer function $U(z)$ as

$$U(z) = \frac{zH(K_p z - K_p + K_i)}{z^2 + z(HK_p - 1) + H(K_i - K_p)}$$

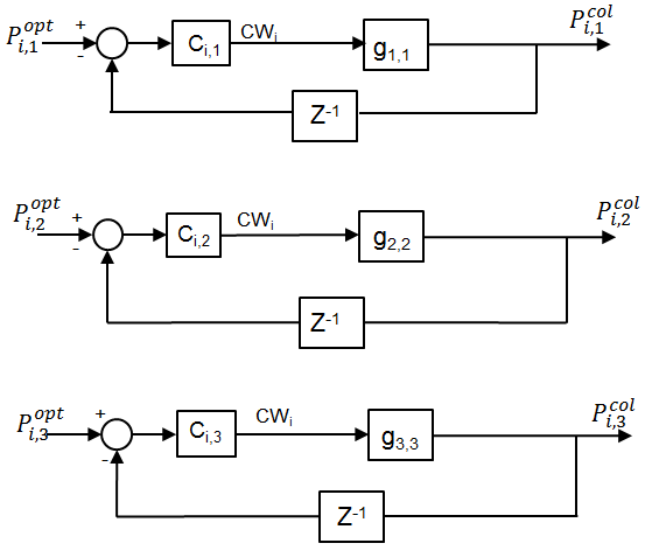

Figure 4.3: Decoupled System

The characteristic equation (Y) for the block diagram is represented as

$$Y = z^2 + z(HK_p - 1) + H(K_i - K_p)$$

and we solve for the controller values using the characteristic second order equation with K_p and K_i given as:

$$K_p = \frac{2\omega_n\xi + 1}{H'(z)}$$

and

$$K_i = \frac{\omega_n^2}{H'(z)} + K_p$$

4.3 Simulation and Performance Evaluation

In this section, we evaluate the proposed algorithm and used simulation parameters shown in Table 4.2 along with physical layer standard values shown in Table 3.1. The first step of the analysis in order to verify the algorithm is to check if the system can indeed be decoupled by using singular value decomposition (svd). This gave a value lower than 50 which shows the system can be decoupled. To meets the objective, experiments for a 2X2 MIMO network was carried out by considering AC_{video} and AC_{BE}. Each AC had the same number of stations transmitting on the network and all stations transmit at the same data rates of 54Mbps. We assume a saturated network and packets of equal size.

Table 4.2: Simulation values

Parameter	Value	Parameter	Value
$\frac{S_{video}}{S_{BE}}$	3.5	T_{data}	$148\mu s$@$54Mbps$
m_{video}	12	m_{BE}	1
d_{video}	$500\mu s$	d_{BE}	$1000\mu s$
λ	1000pkt/sec	%OverShoot	0

Table 4.3: Saturation Scenario Results

n_i	τ_v	τ_d	$H(z)_v$	$H(z)_d$	K_p^{video}	K_i^{video}	K_p^{BE}	K_i^{BE}	CW_v	CW_{BE}
1	0.068	0.303	0.0365	0.1166	4.319	2.879	1.352	0.9013	9	1
2	0.037	0.194	0.0185	0.0580	8.521	5.681	2.718	1.812	20	5
3	0.037	0.188	0.0179	0.0561	8.807	5.871	2.810	1.873	27	8
4	0.033	0.171	0.0155	0.0511	10.171	6.780	3.085	2.057	37	10
5	0.030	0.155	0.0138	0.0468	11.423	7.616	3.368	2.246	45	12
6	0.027	0.142	0.0122	0.0433	12.922	8.614	3.641	2.427	54	14
7	0.025	0.131	0.0111	0.0406	14.202	9.468	3.883	2.589	60	16
8	0.023	0.122	0.0101	0.0384	15.608	10.406	4.105	2.737	70	17
9	0.021	0.114	0.0092	0.0363	17.135	11.423	4.343	2.895	78	20
10	0.020	0.107	0.0086	0.0347	18.331	12.220	4.543	3.029	86	20

Table 4.4: Unsaturation Scenario with ρ=0.5

n_i	τ_v	τ_d	$H(z)_v$	$H(z)_d$	K_p^{video}	K_i^{video}	K_p^{BE}	K_i^{BE}	CW_v	CW_{BE}
1	0.106	0.289	0.0206	0.0153	10.2590	13.6560	13.8131	18.3870	14	7
2	0.041	0.127	0.0111	0.0075	19.0400	25.3442	28.1787	37.5094	24	26
3	0.025	0.082	0.0079	0.0061	26.7519	35.6102	34.6460	46.1118	33	35
4	0.018	0.06	0.0062	0.0053	34.0872	45.3743	39.8755	53.0793	42	43
5	0.014	0.048	0.0051	0.0048	41.4393	55.1609	44.0292	58.6084	51	48
6	0.012	0.040	0.0045	0.0044	46.9645	62.5157	48.0319	63.9365	60	55
7	0.010	0.034	0.0039	0.0041	54.1898	72.1334	51.5464	68.6148	70	68
8	0.009	0.031	0.0174	0.0120	12.1460	16.1678	15.9801	20.3876	35	52
9	0.010	0.033	0.0039	0.0042	54.1898	72.1334	50.3191	66.9811	82	72
10	0.010	0.034	0.0039	0.0044	54.1898	72.1334	48.0319	63.9365	90	78

In figures 4.4, 4.5 and 4.6, we show the network output for the saturated scenario where the throughput allocation ratio is set to 3.5. It is observed that this ratio is maintained on the network and fig 4.5 shows the associated total delay experienced by a TXOP burst. It can be seen that our algorithm maintained the specified condition required to be maintained on the WLAN. We also note from fig 4.5 that as our delay requirement was a bit relaxed initially and as network load increases, the delay limit was reached around the time both traffic types had four nodes on the network. This then gave a network with considerable collision rate and the best effort traffic had low CW values in order to be able to transmit as can be seen in Table 4.3. Fig 4.7 shows the system maintained the desired output at steady state.

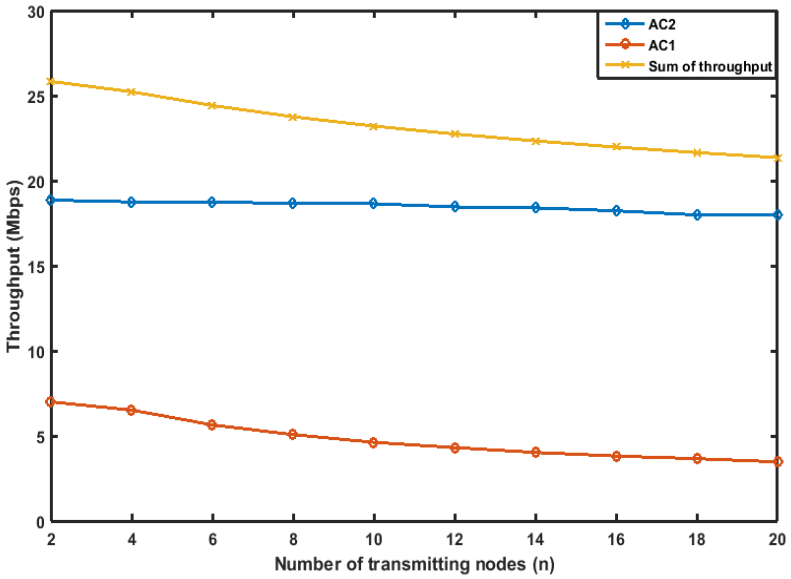

Figure 4.4: Throughput

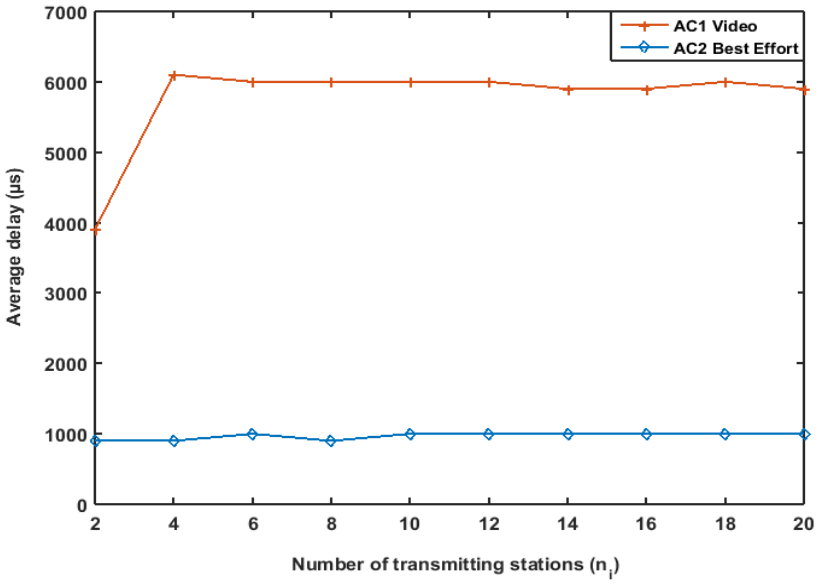

Figure 4.5: Delay

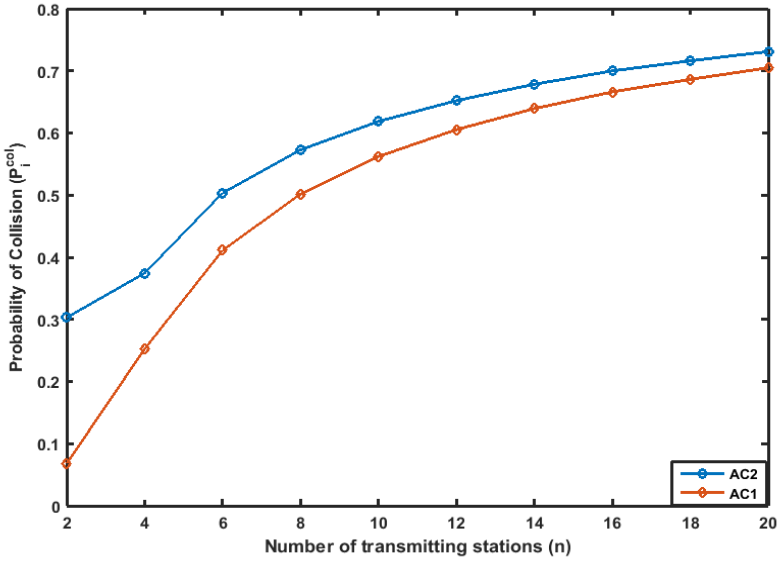

Figure 4.6: Collision Probability

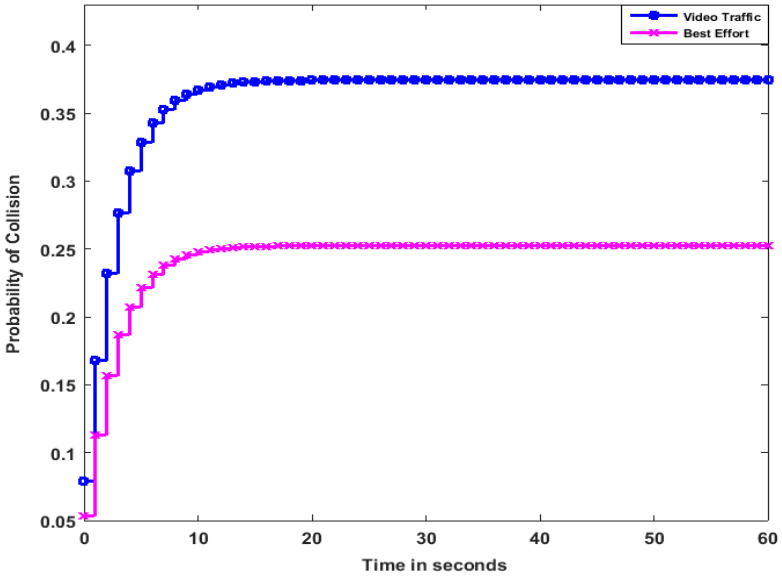

Figure 4.7: System step response

4.3.1 System Stability Analysis

Configuration values for the controllers is a trade off between speed of attaining steady state and stability. From the closed loop transfer function $U(z)$, there are also two zero locations with $z_{0,1} = 0$ being fixed and the tuning of the controller parameters is dependent on location of $z_{0,2}$ which is given as $z_{0,2} = 1 - \frac{K_i}{K_p}$. A positive $z_{0,2}$ location implies $K_p > K_i$ in which the poles z_p are always real and possibility of oscillation is negligible as was seen in 3.1.6.

When $K_p < K_i$, location of $z_{0,2}$ is on the negative axis, and z_p can be real or complex with possibility of system oscillations.

To analyse the stability of the control system, we apply both Jury's and Routh's stability criterion [7] and the system is asymptomatically stable if:

1. $$|H(K_i - K_p)| < 1 \tag{4.15}$$

   From equations 3.23 and 3.24, we have the expression for the proportional and integral controllers. Substituting into equation 4.15, we have that for stability,

   $$\omega_n^2 < 1$$

   and if

   $$\omega_n = \frac{4}{\xi T_s}$$

then the system is stable iff:

$$\xi T_s > 4 \tag{4.16}$$

2. $HK_p - 1$ and $H(K_i - K_p)$ are positive.
   For the case of $HK_p - 1$ being positive, we have that $HK_p = 2\omega_n\xi + 1$ is always greater than 1 as ω_n and ξ are positive variables.
   For the expression $H(K_i - K_p)$ to be positive, K_i must always be greater than K_p. Substituting their respective values, $\frac{\omega_n^2}{H} + K_P$ is greater than K_p when H has a positive value.

Figures 4.8 to 4.14 shows the effects of the stability criterion. In Figure 4.8, we show the root locus diagram for the situation when the poles $z_{p,1}$ and $z_{p,2}$ of the characteristic equation lie on the real axis and the zero $z_{0,2}$ is on the positive real axis. For this location, we have a first order equivalent representation of the system with no oscillations. It can also be noted that the fastest settling time that can be experienced in this situation is $T_s = 4seconds$ due to the stability restraint of equation 4.15. The corresponding frequency plot is shown in figure 4.9.

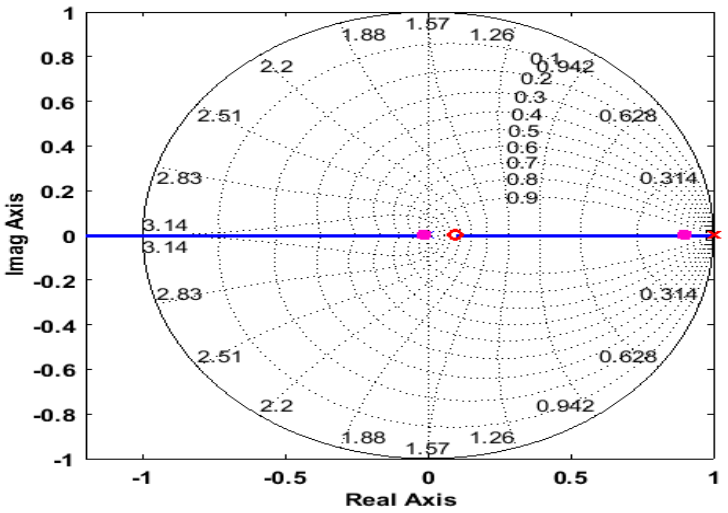

Figure 4.8: Root locus for $z_{0,2}$ on positive real axis

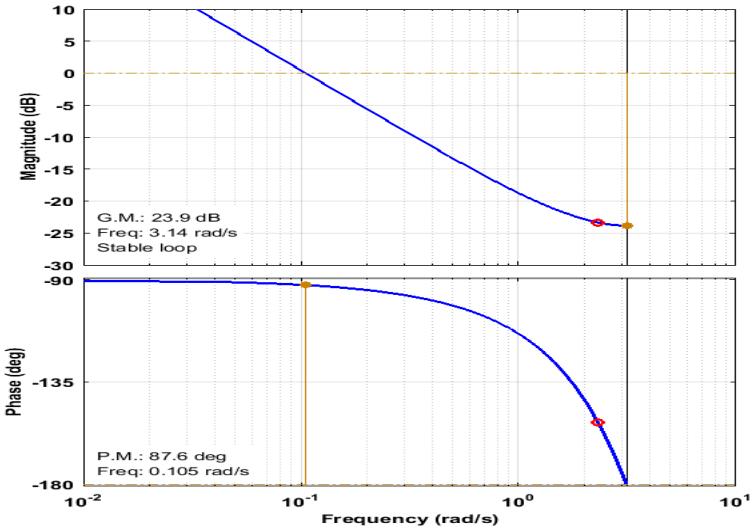

Figure 4.9: Bode plot for $z_{0,2}$ on positive real axis

Figures 4.10 and 4.11 shows the plot for the zero, $z_{0,2}$, on the negative real axis while figures 4.12, 4.13 and 4.14 shows the step response, root locus and Bode plots for the complex root situation where there is an overshoot and settling time $T_s < 4seconds$

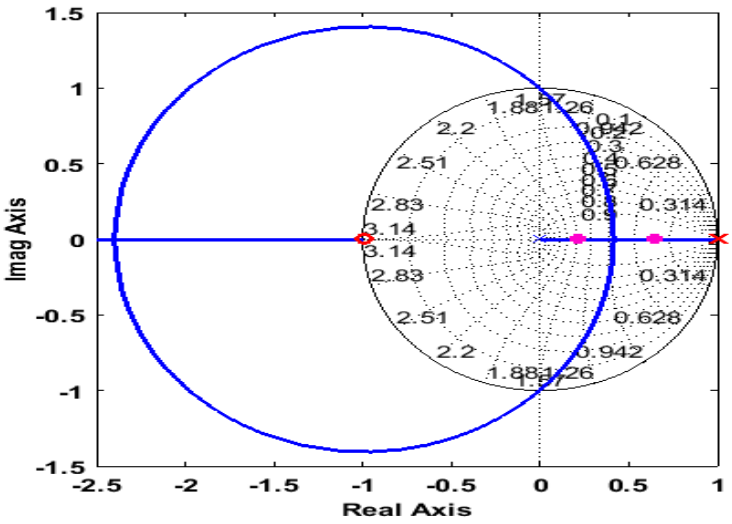

Figure 4.10: Root locus for $z_{0,2}$ on negative real axis

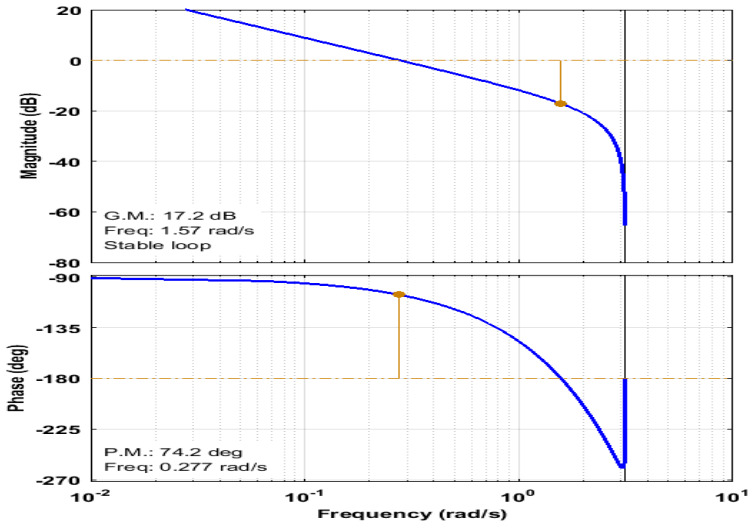

Figure 4.11: Bode plot for $z_{0,2}$ on negative real axis

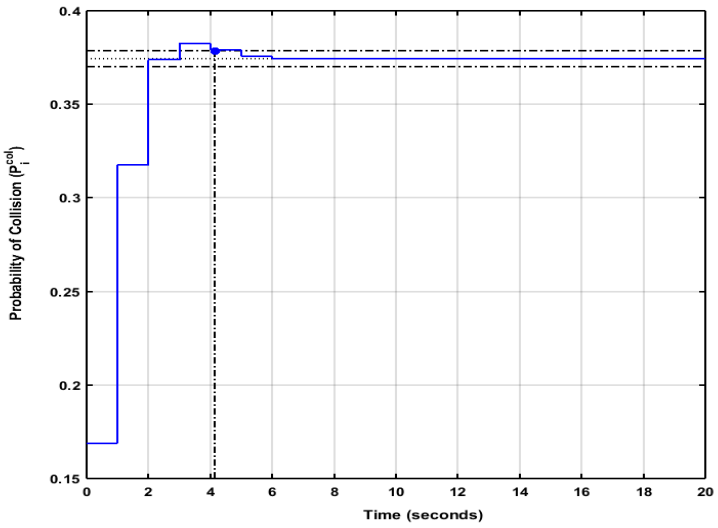

Figure 4.12: Step response for $z_{0,2}$ on negative real axis with complex poles

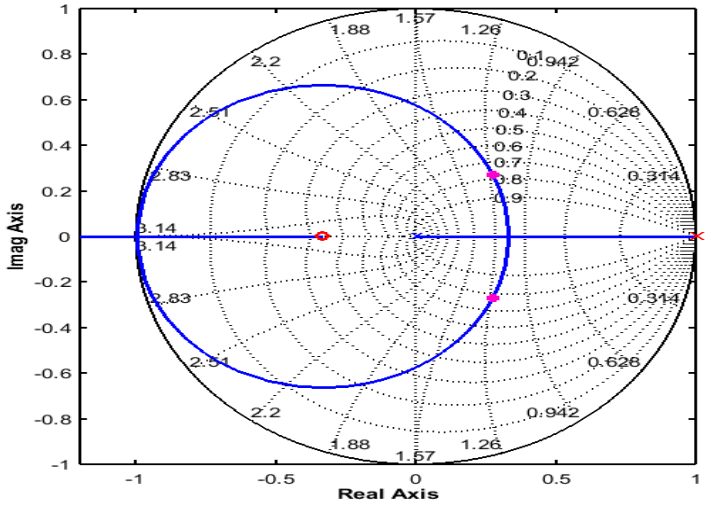

Figure 4.13: Root locus for $z_{0,2}$ on negative real axis with complex poles

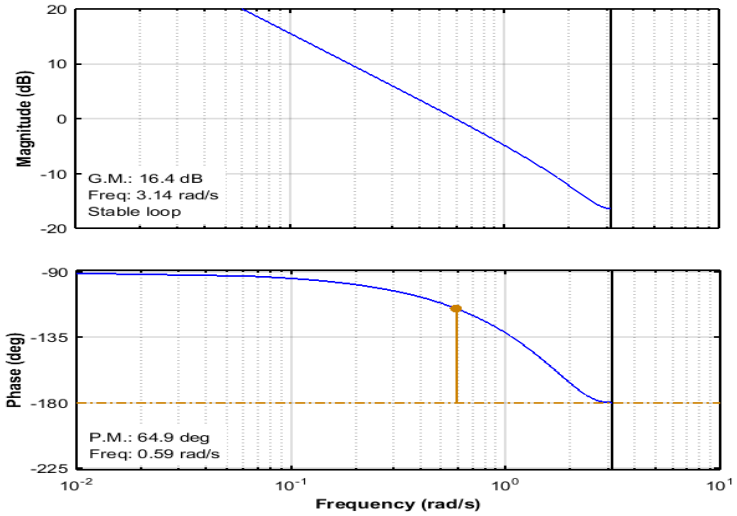

Figure 4.14: Bode for $z_{0,2}$ on negative real axis with complex poles

4.4 Conclusion

In this chapter, we considered a WLAN with stations transmitting more than one traffic type. We also considered delay constraints for the delay sensitive traffic as a QoS requirement and we developed a control algorithm which outputs the optimal value of CW to the contending stations. Our algorithm used a decoupler control mechanism to cancel out interferences and we were able to then control individual components of the Multiple-Input-Multiple-Output (MIMO) network system. The performance of the algorithm was validated by simulations and stability and fast convergence of the controller has been shown both theoretically

and through simulations.

Chapter 5

Decentralised Control of an Error-prone Channel

In the preceding chapter, we considered a wireless network with an ideal channel which implies that losses on the network are due to collisions experienced by the transmitted packets only. For a more realistic scenario, there is need to model the WLAN giving consideration to channel noise where there are transmission losses and varying bit error rates.

In this chapter, we consider an error prone network where additional losses can be incurred due to the channel being noisy. We simulate an error prone network which considers different bit error rates and we derive throughput and delay performance metrics based on this additional assumption. The contribution in this chapter is in the extension of the functionality of the designed decoupled controller in chapter 4 by applying a ceiling to the controller output value giving rise to a control system with system input constraints. The upper and lower limits of the constraints are set using the CW_{min} values as prescribed in the IEEE 802.11e standard and we designed an anti-wind system that counters the effect of the saturation introduced by the constraints.

5.1 Throughput Analysis

In this section, we represent the throughput of the i^{th} AC, (S_i) as:

$$S_i = \frac{P_i^s \mathbb{E}(m_i L)}{P^e T^e + \sum_{i=0}^{N-1} P_i^s T_i^s + (P^B - \sum_{i=0}^{N-1} P_i^s) T^c + \sum_{i=0}^{N-1} P_i^{ec} T_i^{ec}} \tag{5.1}$$

where

L is the size of a single packet
m_i is the number of packets in one $TXOP_i$
$\mathbb{E}(m_iL)$ is the size of payload successfully transmitted in an erroneous channel
N is the number of ACs transmitting on the WLAN
P_i^{ec} is the probability of an error occurring in a slot and is given as

$$P_i^{ec} = n_i\tau_i(1-\tau_i)^{n_i-1}\prod_{\substack{j=0\\j\neq i}}^{N-1}(1-\tau_j)^{n_j}P_i^e$$

The probability of successful transmission of frame of AC_i is given as

$$P_i^s = \tau_i(1-\tau_i)^{n_i-1}(1-P_i^e)\prod_{\substack{j=0\\j\neq i}}^{N-1}(1-\tau_j)^{n_j}$$

where P_i^e is the probability of error in a packet and is given as $1-(1-P^{err})^{m_i}$ and P^{err} is the packet error rate.

In this scenario, a randomly chosen slot time could also contain a failed transmission due to an erroneous channel. We then define the timing of this occurrence as T^{ec} which is the time used in transmitting an erroneous packet.

So, defining all the timings in the throughput expression, we have:

$$\begin{cases} T_i^s = (1-P_b)^{N_i^b}\big(T_i^o + N_i^b(T^{oo}+\frac{L}{C})\big) \\ T_i^{ec} = \sum_{k=1}^{N_i^b}(1-P_b)^{k-1}P_b\big(T_i^o + k(T^{oo}+\frac{L}{C})\big) \\ T^c = T^{rts} + T^{ACK} + SIFS + DIFS \end{cases} \tag{5.2}$$

where $T^{oo} = T^{phy} + 2SIFS + T^{ACK}$ is protocol overhead per packet and $T_i^o = T^{rts} + SIFS + T^{cts} + AIFS_i$ is the protocol overhead with the transmission burst. T^{rts} and T^{cts} is time taken to send the RTS/CTS packets.

τ_i is the probability that AC_i transmits in a randomly chosen slot. We have calculated this parameter as

$$\tau_i = \frac{2(1-P_i^{blk})\alpha_iP_i^p}{2(1-P_i^{blk})\alpha_i(P_i^p+(1-\rho_i)) + P_i^p(CW_i^{mim}K_i-\alpha_i)} \tag{5.3}$$

where
$\alpha_i = (1-2P_i^f)(1-(P_i^f)^{m+1})$ and $K_i = (1-P_i^f)(1-(2P_i^f)^{m+1})$.

P_i^f is the probability of transmission failure of station with AC_i and is given as

$$P_i^f = P_i^c + (1 - P_i^c)P_i^e$$

CW_i^{min}, P_i^p, ρ_i, D_i^T, λ_i are defined in section 5.1.

5.1.1 Delay Analysis

We considered delay as time the HOL frame spends from the beginning of contending for channel access till the last bit of the packet has been successfully sent. Since our approach in this paper is a follow on from [19], we set the range of CW_i^{min} to CW_i^{max} as a singular value CW_i and do not consider number of retransmission in our delay analysis.

Expected Countdown Delay (D_i^{cd})

This stipulates time spent during back-off timer countdown i.e. the average number of idle slots a station counts down during backoff stage without considering times that the backoff counter freezes. For any backoff stage, the average number of slots counted down is given as

$$\begin{aligned} D_i^{cd} =& T^e \Bigg[P_i^p (1 - P_i^b)(1 - \rho_i) \Big(\sum_{j=1}^{M} (P_i^f)^j (1 - P_i^f) \sum_{h=1}^{j} \frac{CW_{i,h}}{2} + (P_i^f)^{M+1} \sum_{h=1}^{M} \frac{CW_{i,h}}{2} \Big) \\ &+ \sum_{j=0}^{M} (P_i^f)^j (1 - P_i^f) \sum_{h=0}^{j} \frac{CW_{i,h}}{2} + (P_i^f)^{M+1} \sum_{h=0}^{M} \frac{CW_{i,h}}{2} \Bigg] \end{aligned}$$

Expected Blocking Delay (D_i^{blk})

Blocking delay is the estimated delay experienced by the frame of AC_i due to freezing its counter when the medium is sensed as busy. This freeze time varies with each AC_i due to their different t_i and also depends on whether freezing the counter was due to an ongoing transmission on the channel or a collision due to transmission attempt by another STA on the WLAN. The estimated delay due to a successful transmission is given as:

$$D_i^{bt} = \frac{(n_i-1)\tau_i}{(1-\tau_i)^2}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}T_i^s + \sum_{j=0,j\neq i}^{N-1}\frac{n_j\tau_j}{(1-\tau_i)(1-\tau_j)}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}T_j^s$$

and the estimated delay due to a collision on the WLAN is:

$$\begin{aligned}D_i^{bc} =& T^c\Bigg[1-(1-\tau_i)^{n_i-1}\prod_{\substack{h=0\\h\neq i}}^{N-1}(1-\tau_h)^{n_h} - \frac{(n_i-1)\tau_i}{(1-\tau_i)^2}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}\\ &+\sum_{\substack{j=0\\j\neq i}}^{N-1}\frac{n_j\tau_j}{(1-\tau_i)(1-\tau_j)}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}\Bigg]\end{aligned}$$

The expected blocking delay experienced by AC_i is:

$$D_i^{blk} = \frac{D_i^{cd}}{T^e}(D_i^{bt} + D_i^{bc}) \tag{5.4}$$

Retransmission Delay (D_i^{col})

For a STA transmitting, one of the possibilities is retranx. The delay experienced from retransmission is a combination of collision delay and delay due to erroneous transmission. These are estimated as follows:

$$\begin{aligned}D_i^{col} =& T^c\Bigg[\sum_{j=0}^{m} j(P_i^c)^j(1-P_i^c) + (m+1)(P_i^c)^{m+1} + P_i^p(1-\rho_i)(1-P^B)\\ &\left(\sum_{j=1}^{m} j(P_i^c)^j(1-P_i^c) + (m+1)(P_i^c)^{m+1}\right)\Bigg]\end{aligned}$$

Transmission Delay (D_i^{trans})

The transmission delay is time taken to successful transmit a frame multiplied by the probability that the frame does not fail

$$D_i^{trans} = \mathbb{E}(N_b L)\Bigg[(1-P_i^e)\sum_{j=0}^{m} j(1-P_i^c)^j (P_i^e)^{j-1} + (m+1)(1-P_i^c)^{m+1}(P_i^e)^{m+1} + P_i^p(1-\rho_i)\Bigg((1-P_i^c)\sum_{j=1}^{M} j(P_i^c)^j + (M+1)(P_i^c)^{M+1}\Bigg)\Bigg]$$

The average total delay D_i^T experienced by the HOL frame is then calculated as the sum of the different components analysed above, which gives:

$$D_i^T = \frac{D_i^{cd} + D_i^{blk} + D_i^{col} + D_i^{trans}}{\sum_{k=1}^{N_i^b} k(1-P_b)^{k-1}P_b + N_i^b(1-P_b)^{N_i^b}}$$

5.1.2 WLAN transfer function

The WLAN transfer function $G(z)$ is the linear relationship between the input and output of the system. From equations (5.3) and (4.7), we see that the relationship between CW_i and P_i^f is non-linear so we derive a linear approximation of $G(z)$ by linearising about the stable point of interest P_i^f. This implies:

$$G(z) = \begin{cases} \frac{\partial P_i^f}{\partial CW_i} = \sum_{k=0}^{N-1} \frac{\partial P_i^f}{\partial \tau_k}\frac{\partial \tau_k}{\partial CW_i} \\ \\ \frac{\partial P_i^f}{\partial CW_j} = \sum_{k=0}^{N-1} \frac{\partial P_i^f}{\partial \tau_k}\frac{\partial \tau_k}{\partial CW_j} \end{cases} \tag{5.5}$$

where

$$\frac{\partial P_i^f}{\partial \tau_i} = \frac{n_i - 1}{(1-\tau_i)^2}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}(1-P_i^{ec})$$

$$\frac{\partial P_i^f}{\partial \tau_j} = \frac{n_j}{(1-\tau_i)(1-\tau_j)}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}(1-P_i^{ec})$$

and

$$\frac{\partial \tau_i}{\partial CW_i} = \frac{1}{K_i P_i^p \tau}\Bigg[2\alpha_i(1-P_i^b)\Big((1-\tau_i)X_i - P_i^p - (1-\rho_i)\Big) + \Big(P_i^p(1-\tau_i) - \tau_i(1-\rho_i)\Big)$$
$$\Big(\alpha_i V_i + 2Z_i(1-P_i^b)\Big) + P_i^p(\tau_i Z_i + \alpha_i) + \tau_i\alpha_i Z_i\Bigg] - \frac{1}{(K_i P_i^p \tau)^2}\Bigg[\Big(P_i^p\tau_i Y_i + K_i(X_i\tau_i + P_i^p)\Big)$$
$$\Big(2\alpha_i(1-P_i^b)(P_i^p(1-\tau_i) - \tau_i(1-\rho_i) + P_i^p\tau_i\alpha_i)\Big)\Bigg]$$

$$\frac{\partial \tau_i}{\partial CW_j} = \frac{1}{K_j P_j^p \tau_j}\Bigg[X_j\alpha_j\Big(2(1-P_j^b)(1-\tau_j) + \tau_j\Big) + \Big(V_j\alpha_j + Z_j 2(1-P_j^b)\Big)\Big(P_j^p(1-\tau_j) - \tau_j$$
$$(1-\rho_j)\Big) + \tau_j P_j^p Z_j\Bigg] - \frac{1}{(K_j P_j^p \tau_j)^2}\Bigg[\Big(\tau_j(P_j^p Y_j + K_j X_j)\Big)\Big(2\alpha_j(1-P_j^b)(P_j^p(1-\tau_j) -$$
$$\tau_j(1-\rho_j)\Big) + P_j^p\tau_j\alpha_j)\Big)\Bigg]$$

where V denotes the derivative of $(1-P_i^b)$ and this is given as

$$\begin{cases} V_i = -2(1-P_i^b)(A_i - A_{min} + 1)^{\frac{n_i-1}{1-\tau_i}} \\ V_j = -2(1-P_j^b)(A_j - A_{min} + 1)^{\frac{n_i}{1-\tau_i}} \end{cases}$$

X_i and X_j are partial derivatives of P_i^P with respect to τ_i and τ_j respectively and are given as:

$$X_i = \lambda_i e^{-\lambda_i T^{cs}} X$$
$$X_j = \lambda_j e^{-\lambda_j T^{cs}} X$$

X is the derivative of T^{cs} and is given as

$$X = \frac{n_i}{(1-\tau_i)}\prod_{h=0}^{N-1}(1-\tau_h)^{n_h}(T^c - T^e) + \Big(n_i(1-\tau_i)^{n_i-1} - n_i\tau_i(n_i-1)(1-\tau_i)^{n_i-2}\Big)$$
$$\prod_{\substack{h=0\\h\neq i}}^{N-1}(1-\tau_h)^{n_h}(T_i^s - T^c) - \sum_{\substack{h=0\\h\neq i}}^{N-1}\frac{(n_i n_j)\tau_j}{(1-\tau_i)(1-\tau_j)}\prod_{k=0}^{N-1}(1-\tau_k)^{n_k}(T_i^s - T^c)$$

We note that with the derivatives, when AC_j is of lesser priority compared to AC_i, the number of flows reduces by one and this is implemented accordingly.

Y denotes the derivative of K_i and this is given as

$$\begin{cases} Y_i = \big((2P_i^f)^{m+1} - 2(m+1)(2P_i^f)^m(1-P_i^f) - 1\big)\frac{\partial P_i^f}{\partial \tau_i} \\ Y_j = \big((2P_j^f)^{m+1} - 2(m+1)(2P_j^f)^m(1-P_j^f) - 1\big)\frac{\partial P_j^f}{\partial \tau_i} \end{cases}$$

and Z denotes the derivative of α which is given by:

$$\begin{cases} Z_i = \big(2(P_i^f)^{m+1} - (m+1)(1-2P_i^f)(P_i^f)^m - 2\big)\frac{\partial P_i^f}{\partial \tau_i} \\ Z_j = \big(2(P_j^f)^{m+1} - (m+1)(1-2P_j^f)(P_j^f)^m - 2\big)\frac{\partial P_j^f}{\partial \tau_i} \end{cases}$$

From equation 5.5, we obtain a linearised multivariable transfer function matrix of the WLAN which is shown in equation (5.6) and represented in fig 4.2.

$$G(z) = \begin{pmatrix} \frac{\partial P_0^{col}}{\partial CW_0} & \frac{\partial P_0^{col}}{\partial CW_1} & \cdots & \frac{\partial P_0^{col}}{\partial CW_{N-1}} \\ \frac{\partial P_1^{col}}{\partial CW_0} & \frac{\partial P_1^{col}}{\partial CW_1} & \cdots & \frac{\partial P_1^{col}}{\partial CW_{N-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_{N-1}^{col}}{\partial CW_0} & \frac{\partial P_{N-1}^{col}}{\partial CW_1} & \cdots & \frac{\partial P_{N-1}^{col}}{\partial CW_{N-1}} \end{pmatrix} \tag{5.6}$$

5.1.3 Controller configuration and Stability

The next task is to obtain the controller configuration. For the feedback control loop to be stable and without steady state errors, the designed controller must include an integrator which is the reason for the choice of a PI controller.

The controller matrix is of the form:

$$C(z) = \begin{pmatrix} c_{i,i} & 0 & \cdots & 0 \\ 0 & c_{i,i} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_{i,i} \end{pmatrix}$$

where $c_{i,i}$ has a general expression given as:

$$c_{i,i} = K_p + \frac{K_i}{z-1}$$

Now, the closed loop transfer function $U(z)$ is obtained as:

$$U(z) = G(z)D(z)C(z)$$

$$U(z) = \frac{zH(K_p z - K_p + K_i)}{z^2 + z(HK_p - 1) + H(K_i - K_p)} \tag{5.7}$$

and following this the controller parameters are defined as

$$K_p = \frac{2\omega_n \zeta + 1}{H} \tag{5.8}$$

$$K_i = K_p + \frac{\omega_n^2}{H} \tag{5.9}$$

where ω_n and ζ are tuning parameters representing the undamped natural frequency of the system which it gives the speed of response of the system and damping factor.

Configuration values for the controllers is a trade off between speed of attaining steady state and stability. From the closed loop transfer function $U(z)$, we note that there are two zero locations - $z_{0,1} = 0$ which is fixed and $z_{0,2} = 1 - \frac{K_i}{K_p}$. A positive $z_{0,2}$ location implies $K_p > K_i$ in which the poles z_{p1} and z_{p2} are always real and possibility of oscillation is negligible as system has first order characteristics.

When $K_p < K_i$, location of $z_{0,2}$ is on the negative axis, and z_{p1} and z_{p2} can be real or complex with possibility of system oscillations.

To analyse the stability of the control system, we note first that the roots of the equation (5.7) must lie within the unit circle which implies that:

$$H(K_i - K_p) < 1 \tag{5.10}$$

Applying Jury's stability criterion [7], the system is asymptomatically stable if the following inequalities hold:

$$HK_i > 0 \tag{5.11}$$

$$H\left(K_i - 2K_p\right) > -2 \tag{5.12}$$

5.1.4 Simulation and Performance Evaluation

In this section, we evaluate our proposed control algorithm through simulations using the control toolbox simulator in Matlab R2015a and the performance of the algorithm was verified and compared with the standard EDCA protocol using OMNeT++ simulator.

First, we designed the FCA controller and verified the stability of the controller using Matlab. We also obtained the CW values which was then implemented on the WLAN using the OMNeT++ simulator. For the controller design, we assumed a zero percent overshoot and a settling time of 20 seconds as shown in Table 3.1.

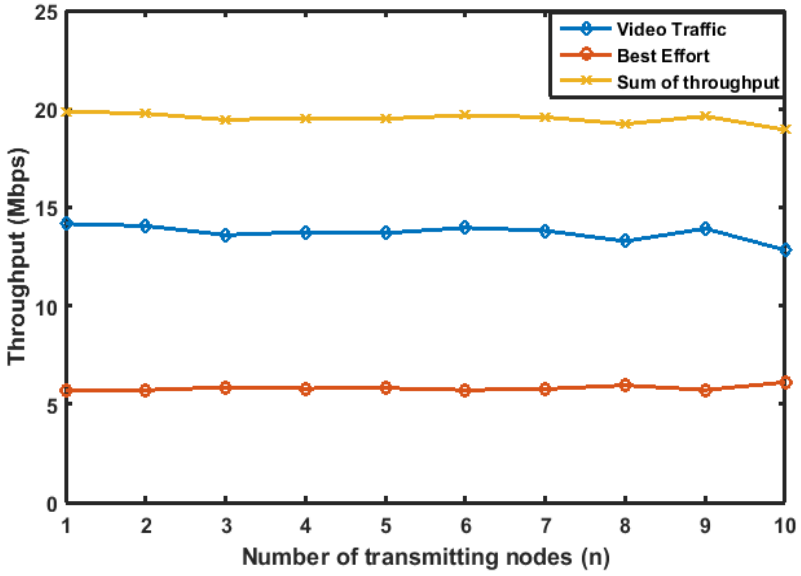

Figure 5.1: Sum of Throughput for $ber = 10^{-4}$

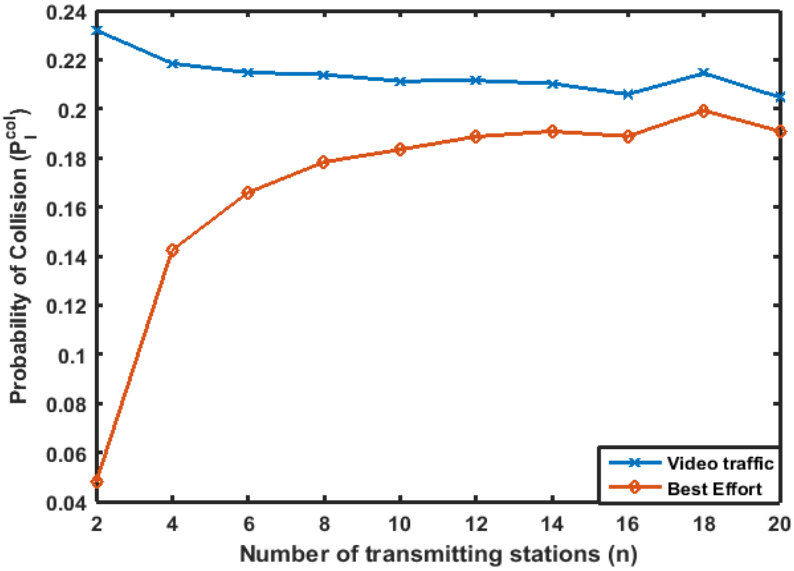

Figure 5.2: Probability of Collision for $ber = 10^{-4}$

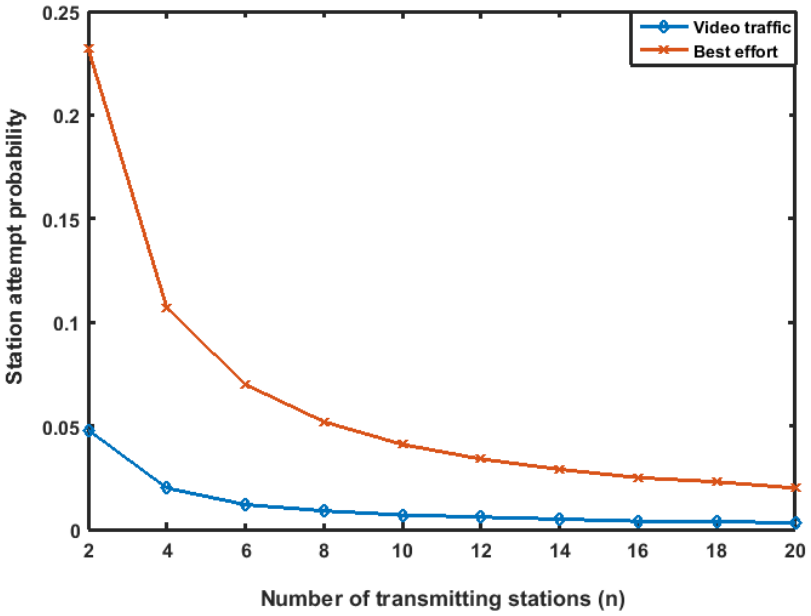

Figure 5.3: Station attempt probability for $ber = 10^{-4}$

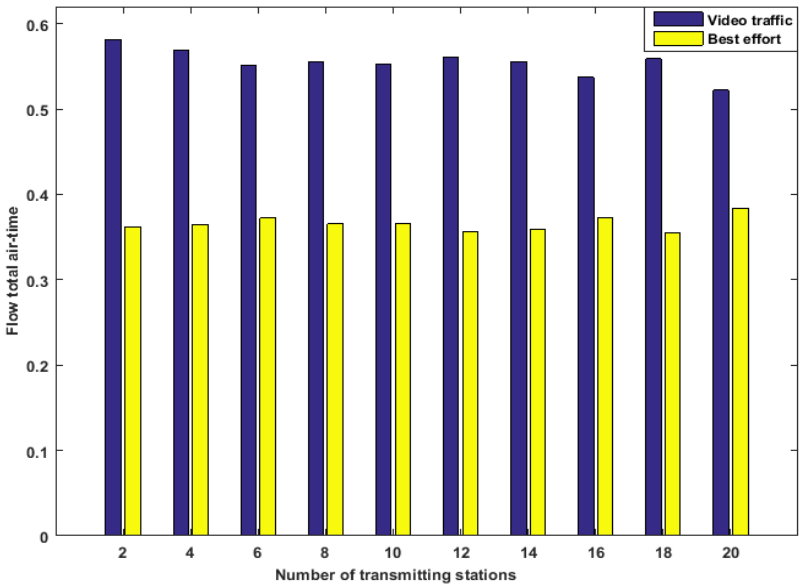

Figure 5.4: Flow total air-time

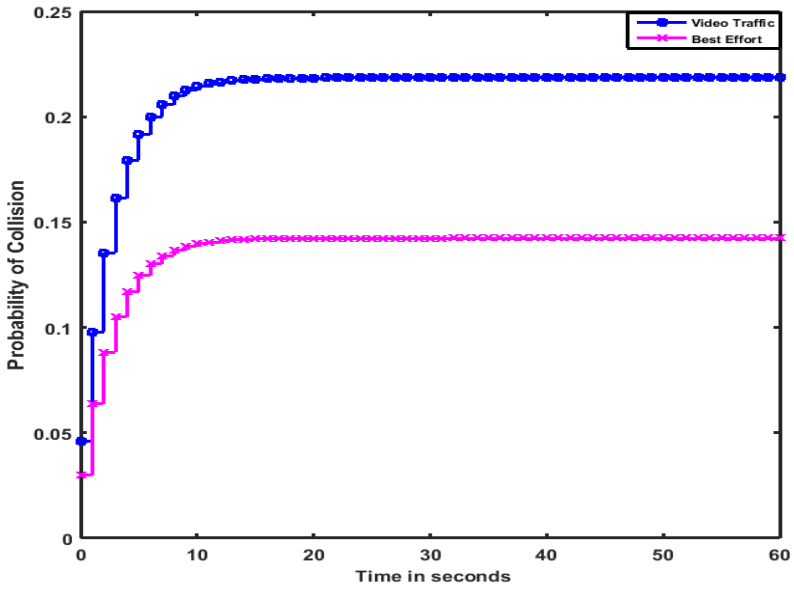

Figure 5.5: System step response

5.2 Constrained Controller Configuration and Stability

One of the aims of this research work is devicing an alternative algorithm that achieves the EDCA protocol functionalities so, we apply some system input constraints in order to keep a boundary on the CW values sent as inputs to the network. The block diagram layout for this is shown in figure 5.6.

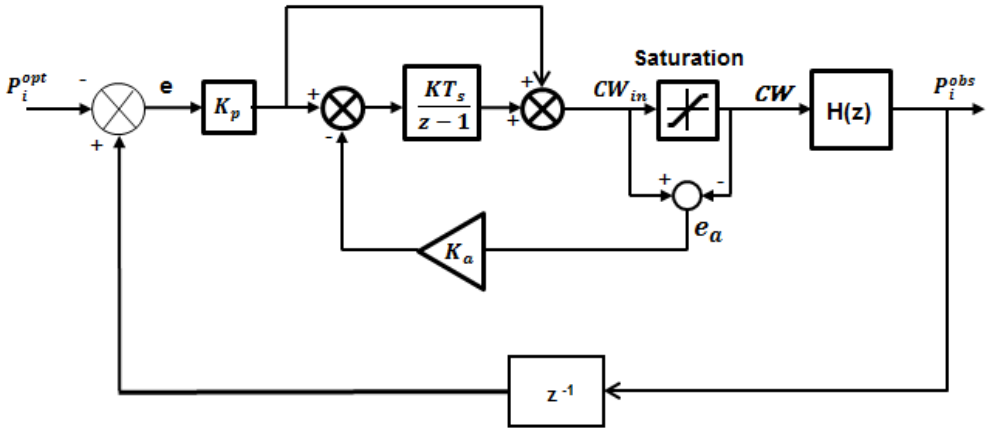

Figure 5.6: Anti-windup Control

The introduction of saturation or constraints in a control system brings instability and control functions are interrupted. When the controller outputs exceeds the stipulated limits, "windup" occurs in the controller and an anti-windup loop is included in order to counter the windup effects and restore stability to the system. In figure 5.6, stability of the system is achieved through the gain K_a.

Considering the anti-windup loop, characteristic equation is given as:

$$1 + \frac{K_a K T_s}{z - 1} = 0 \tag{5.13}$$

where $K = \frac{K_i}{K_p}$. This implies

$$z - 1 + K_a K T_s = 0$$

Using Routh-Hurwitz stability criterion and the bilateral transformation $z = \frac{w+1}{w-1}$ in order to derive the stability range of K_a, we have

$$w + 1 - 1(w - 1) + K_a K T_s (w - 1) = 0$$

$$w + \frac{2}{K_a K T_s} - 1 = 0$$

$$w + \frac{2 - K_a K T_s}{K_a K T_s} = 0$$

which implies that for stability,

$$0 < K_a K T_s \leq 2$$

K and T_s are already known design parameters so it is easy to compute the range of K_a. The values that K_a takes is in a range and the range affects the transient response of the controller with the lower values yielding a slower response and the pace of response getting faster as the value of K_a increases.

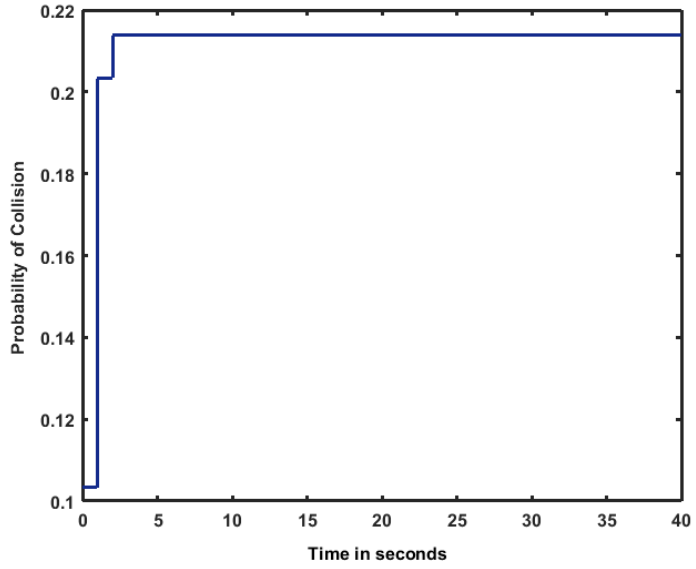

Figure 5.7: System step response for $K_a = 1.5$

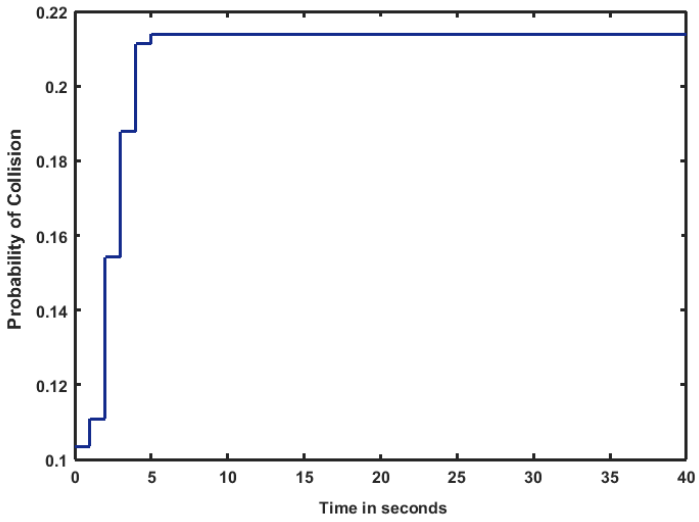

Figure 5.8: System step response for $K_a = 0.05$

In figure 5.7, we show the system response for $K_a = 1.5$ which is the maximum limit when $T_s = 1$second and $K = 1.331$. Figure 5.8 shows a lower value of $K_a = 0.005$ which is closer to the lower limit. It can be seen in the two outputs that the controller response is faster as K_a increases.

5.3 Conclusion

In this chapter, we considered a WLAN with stations transmitting more than one traffic type with particular QoS requirements and developed a control al-

gorithm which outputs the optimal value of CW to the contending stations. Our algorithm consists of a PI controller and introducing a decoupler was necessary as the conditions of the WLAN depicts a Multiple-Input-Multiple-Output (MIMO) system. The decoupler cancels out interferences which enables us to treat each AC_i as a Single-Input-Single-Output (SISO) system and give accurate outputs. The performance of the algorithm was validated by simulations and stability and fast convergence of the controller has been shown both theoretically and through simulations.

Chapter 6

Optimal Control Network

In this chapter, we follow on with the analysis is chapter 4. We maintain the throughput definitions and the average slot timings and their corresponding definitions. We use the RTS/CTS mode of transmission and considered a saturated network. The distinctive difference here is that while we still consider a network with multiple ACs, we allocate network resources using the Proportional fairness criteria and used the Linear-Quadratic-Integral Control for the controller algorithm.

The throughput of a station carrying a flow of AC i is thus given by

$$s_i(\boldsymbol{\tau}) = \frac{P_i^{succ} m_i L}{P^e T^e + \sum_{i=0}^{N-1} P_i^{succ} T_i^{succ} + (1 - P^e - P^{succ}) T^{col}}$$

By working in terms of the quantity $\alpha_i = \frac{\tau_i}{1-\tau_i}, \alpha_i > 0$ instead of τ_i, the station throughput is rewritten as

$$s_i = \frac{\alpha_i m_i L}{X \cdot T^{col}} \tag{6.1}$$

in which

$$X = \frac{\sigma}{T^{col}} + \sum_{i=0}^{N-1} n_i \Big(\frac{T_i^{succ}}{T^{col}} - 1 \Big) \alpha_i + \prod_{i=0}^{N-1} \big(1 + \alpha_i\big)^{n_i} - 1$$

According to the throughput model in [36], when $CW_{max} = CW_{min}$ the station attempt probability under saturation conditions can be reduced to

$$\tau_i = \frac{2(1 - P_i^{blk})}{2(1 - P_i^{blk}) + W_i - 1} \tag{6.2}$$

6.1 Average delay

Next, we will calculate the average delay experienced by a TXOP burst of each AC. We start with the analysis for ordinary 802.11 MAC scheduling with binary exponential backoff algorithm and then move onto the scenario when $CW_{min} =$

CW_{max}.

The delay is defined in this work as the duration since a station starts contending for the medium until the transmission is finished (either received successfully or dropped because of reaching the maximum retry limit). The calculation is based on the EDCA WLAN throughput model derived in [36]. The average delay consists of four expected delays, described as follows:

- *Expected countdown delay*: For each backoff stage j ($0 \leq j \leq M$, and M is the retry limit. We assume that $CW^i_{max} \geq 2^M CW^i_{min}$), the average countdown delay for AC i is $CW_{i,j}\sigma/2$, in which $CW_{i,j} = 2^j CW^i_{min}$ is the contention window at the jth backoff stage. The expected delay associated with the backoff countdown process is then given by

$$D_i^{cd} = \sigma \times \left(\sum_{j=0}^{M} (P_i^{col})^j (1 - P_i^{col}) \sum_{h=0}^{j} \frac{CW_{i,h}}{2} + (P_i^{col})^{M+1} \sum_{h=0}^{M} \frac{CW_{i,h}}{2} \right)$$

  where P_i^{col} is the conditional collision probability for the ith AC in the throughput model, i.e.

$$P_i^{col} = 1 - (1 - \tau_i)^{n_i - 1} \prod_{\substack{j=0 \\ j \neq i}}^{N-1} (1 - \tau_j)^{n_j} \tag{6.3}$$

- *Expected blocking delay*: During the countdown process, when a transmission is detected on the channel the backoff time counter is "frozen", and reactivated again after the channel is sensed idle for a certain period. A station is called blocked in our analysis when it senses (an) ongoing transmission(s) from some other station(s) during its countdown process. The blocking delay is the period during which a station is "frozen". The expected number of time slots in the backoff countdown process is D_i^{cd}/σ. At each time slot, a station could be blocked by either a successful transmission or a collision. For a station of AC i, the delay caused by a successful transmission from some other station is

$$D_i^{bs} = T_i^{succ}(n_i - 1)\tau_i(1 - \tau_i)^{n_i - 2} \prod_{\substack{j=0 \\ j \neq i}}^{N-1} (1 - \tau_j)^{n_j} + \sum_{\substack{j=0 \\ j \neq i}}^{N-1} T_j^{succ} n_j \tau_j (1 - \tau_j)^{n_j - 1}$$

$$\prod_{\substack{k=0 \\ k \neq j,i}}^{N-1} (1 - \tau_k)^{n_k} (1 - \tau_i)^{n_i - 1}$$

The blocking delay because of a collision is

$$D_i^{bc} = T^{col}\Bigg(1-(1-\tau_i)^{n_i-1}\prod_{\substack{j=0\\j\neq i}}^{N-1}(1-\tau_j)^{n_j}-(n_i-1)\tau_i(1-\tau_i)^{n_i-2}\prod_{\substack{j=0\\j\neq i}}^{N-1}(1-\tau_j)^{n_j}$$
$$-\sum_{\substack{j=0\\j\neq i}}^{N-1}n_j\tau_j(1-\tau_j)^{n_j-1}\prod_{\substack{k=0\\k\neq j,i}}^{N-1}(1-\tau_k)^{n_k}(1-\tau_i)^{n_i-1}\Bigg)$$

The expected blocking delay of AC i is thus

$$D_i^{blk} = \frac{D_i^{cd}}{\sigma}(D_i^{bs}+D_i^{bc})$$

- *Expected retransmission delay*: The expected retransmission delay for AC i is calculated by multiplying the expected number of retransmission attempts by the collision duration, i.e.

$$D_i^{retx} = T^{col}\times$$
$$\Bigg(\sum_{j=0}^{M} j(P_i^{col})^j(1-P_i^{col})+(M+1)(P_i^{col})^{M+1}\Bigg)$$

- *Expected successful transmission delay*: The expected successful transmission delay is the duration of a successful transmission multiplied by the probability that the transmission is not dropped, which is given by

$$D_i^{succ} = T_i^{succ}(1-(P_i^{col})^{M+1})$$

Combining the above four delays, the average delay of a TXOP burst of AC i is therefore given by

$$D_i = D_i^{cd}+D_i^{blk}+D_i^{retx}+D_i^{succ}$$

The proposed approach in this paper works by finding the optimal contention window to achieve proportional fairness amongst ACs , so the exponential backoff algorithm is unnecessary in our setting and we simply set $CW_{max}^i = CW_{min}^i$, i.e. $M = 0$. To simplify notations, we hereafter refer to CW_{min}^i with W_i. The four expected delays then become:

$$D_i^{cd} = \sigma\frac{W_i}{2}$$

$$D_i^{blk} = \frac{W_i}{2}(D_i^{bs}+D_i^{bc})$$

$$D_i^{retx} = T^{col}P_i^{col}$$

$$D_i^{succ} = T_i^{succ}(1 - P_i^{col})$$

Similarly by working in terms of the quantity $\alpha_i = \frac{\tau_i}{1-\tau_i}$, the average delay experienced by a TXOP burst of AC i when $M = 0$ is then

$$\begin{aligned} D_i = &\frac{W_i(\sigma + T^{col})}{2} + \frac{Y_i}{(1+\alpha_i)^{n_i-1}}(T_i^{succ} - T^{col}) + T^{col} \\ &+ \frac{W_i Y_i}{2(1+\alpha_i)^{n_i-1}}\Big(Z_i - T^{col} + (T_i^{succ} - T^{col})(n_i - 1)\alpha_i\Big) \end{aligned} \tag{6.4}$$

in which

$$Y_i = \sum_{j=0, j\neq i}^{N-1} (1+\alpha_j)^{-n_j}$$

$$Z_i = \sum_{j=0, j\neq i}^{N-1} (T_j^{succ} - T^{col}) n_j \alpha_j$$

and

$$W_i = \frac{2}{\alpha_i}\Big((1+\alpha_i)\prod_{j=0}^{N-1}(1+\alpha_j)^{-n_j}\Big)^{t_i - t_{min}+1} + 1$$

6.2 Proportional fair allocation

The 802.11e EDCA standard provides service differentiation by assigning different contention parameters to distinct ACs. Delay-sensitive traffic flows, such as voice over WLANs and streaming multimedia, are assigned with higher priorities. This mechanism has a significant cost for lower priority traffic flows as they can practically starve in dense network deployment. In this section we aim at finding the optimal $\boldsymbol{\alpha} := [\alpha_i]_{i\in\{0,1,\cdots,N-1\}}$ to achieve fair allocation of station throughputs amongst ACs. Meanwhile we take into account the delay constraints for each AC. The utility function is defined as the sum of the log of station throughputs

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & U(\boldsymbol{\alpha}) := \sum_{i=0}^{N-1} n_i \log s_i(\boldsymbol{\alpha}) \\ \text{s. t.} \quad & D_i(\boldsymbol{\alpha}) \leq m_i d_i \quad 0 \leq i \leq N-1, \\ & \alpha_i > 0 \qquad\qquad 0 \leq i \leq N-1. \end{aligned}$$

in which the station throughput is given by Eqn. (6.1); the average delay is given by Eqn. (6.4); d_i is the delay deadline for a single packet in a TXOP burst of AC i.

By plugging in the station throughput expression and removing the constant

terms, the optimisation problem is simplified as

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & U'(\boldsymbol{\alpha}) := \sum_{i=0}^{N-1} n_i(\log \alpha_i - \log X) \\ \text{s. t.} \quad & D_i(\boldsymbol{\alpha}) \leq m_i d_i \quad 0 \leq i \leq N-1, \\ & \alpha_i > 0 \qquad\qquad 0 \leq i \leq N-1. \end{aligned}$$

6.3 Centralised closed-loop control approach

In this section, we design a centralized adaptive control approach to implement the desirable proportional fairness in real networks. Based upon equation 6.1, the analysis in Section 6.1 and Section 6.2, the proportional fairness is achieved when the station attempt probability parameter $\boldsymbol{\alpha}$ reaches its optimum value $\boldsymbol{\alpha}^*$. The variable $\boldsymbol{\alpha}$ is only determined by the minimum contention window W_i with $AIFS$ and $TXOP$ taking the recommended values and $CW_{max} = CW_{min}$. Our approach uses a multivariable closed-loop control system to tune $\boldsymbol{W}$ to drive the station attempt probability to its optimum. As the station attempt probability is hard to measure in real networks, we measure the conditional collision probability $\boldsymbol{p}^{\boldsymbol{col}}(\boldsymbol{\tau})$ instead of $\boldsymbol{\alpha}(\boldsymbol{\tau})$ in the proposed control approach.

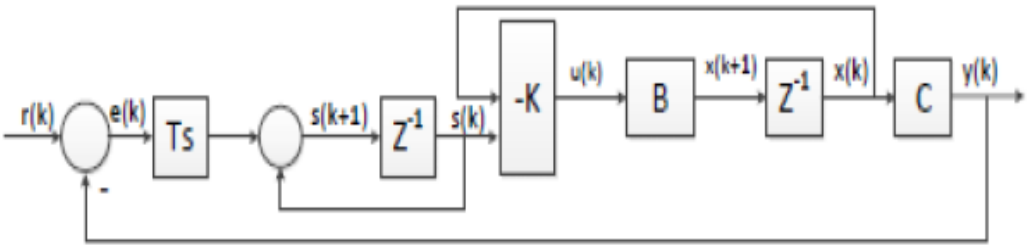

Figure 6.1: LQI controller

The plant is the WLAN itself. The input of the plant is the contention window $\boldsymbol{W} = [W_0, \cdots, W_{N-1}]$, and the output is the observed conditional collision probability $\boldsymbol{p}^{\boldsymbol{obs}} = [p_0^{obs}, \cdots, p_{N-1}^{obs}]$. The design objective is to obtain a stable system in closed-loop with desired performances and shape the output of the system to the given reference value. The reference value is the optimal conditional collision probability $\boldsymbol{p}^*$ given by Eqn. (6.3) for $\boldsymbol{\tau} = \boldsymbol{\tau}^*$.

6.3.1 Linearisation of the non-linear plant

As the proposed adaptive control algorithm is executed every beacon interval, the period is long enough to assume that the measurement corresponds to stationary conditions. Following from previous chapters, we linearise the system about its stable point of operation. Linearisation yields an accurate model of our system about the point of linearisation. Outside of the immediate linearised locality, it will be difficult to guarantee an accurate model and the stability of the proposed model could be affected.

For linearising the system we have that

$$\boldsymbol{H} = \begin{pmatrix} \frac{\partial p_0^{obs}}{\partial W_0} & \frac{\partial p_1^{obs}}{\partial W_0} & \cdots & \frac{\partial p_{N-1}^{obs}}{\partial W_0} \\ \frac{\partial p_0^{obs}}{\partial W_1} & \frac{\partial p_1^{obs}}{\partial W_1} & \cdots & \frac{\partial p_{N-1}^{obs}}{\partial W_1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial p_0^{obs}}{\partial W_{N-1}} & \frac{\partial p_1^{obs}}{\partial W_{N-1}} & \cdots & \frac{\partial p_{N-1}^{obs}}{\partial W_{N-1}} \end{pmatrix}$$

The partial derivatives can be respectively calculated as

$$\frac{\partial p_i^{obs}}{\partial W_i} = \sum_{k=0}^{N-1} \frac{\partial p_i^{obs}}{\partial \tau_k} \cdot \frac{\partial \tau_k}{\partial W_i}$$

and

$$\frac{\partial p_i^{obs}}{\partial W_j} = \sum_{k=0}^{N-1} \frac{\partial p_i^{obs}}{\partial \tau_k} \cdot \frac{\partial \tau_k}{\partial W_j}$$

in which

$$\frac{\partial p_i^{obs}}{\partial \tau_i} = \prod_{k=0}^{N-1} (1-\tau_k)^{n_k} \frac{n_i - 1}{(1-\tau_i)^2}$$

$$\frac{\partial p_i^{obs}}{\partial \tau_j} = \prod_{k=0}^{N-1} (1-\tau_k)^{n_k} \frac{n_j}{(1-\tau_i)(1-\tau_j)}$$

$$\frac{\partial \tau_i}{\partial W_i} = \frac{\tau_i^2}{-2\big(1-P_i^{blk}\big)\big(1+(n_i-1)(t_i-t_{min}+1)\tau_i\big)}$$

$$\frac{\partial \tau_i}{\partial W_j} = \frac{\tau_j(1-\tau_i)}{-2\big(1-P_j^{blk}\big)n_i(1-\tau_j)(t_j-t_{min}+1)}$$

At the stable point of operation, $\tau_i = \tau_i^*$, the non-linear plant is thus linearised as

$$\boldsymbol{p^{obs}} = \boldsymbol{W} \cdot \boldsymbol{H}(\boldsymbol{\tau^*}) - \boldsymbol{W^*} \cdot \boldsymbol{H}(\boldsymbol{\tau^*}) + \boldsymbol{p^*} \tag{6.5}$$

6.3.2 State feedback control

With the lumbarisation, the WLAN can be represented as a discrete MIMO LTI state-space model. According to the proposed adaptive algorithm, the conditional collision probability at instant $k+1$ is determined by the contention window input to the WLAN at instant k, the state and measurement equations are therefore given by

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{B}\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \end{cases}$$

in which the system state is the conditional collision probability,

$$\boldsymbol{x}(k) = [\boldsymbol{p}^{\boldsymbol{obs}}(k)]^T$$

the system input is the minimum contention window,

$$\boldsymbol{u}(k) = [\boldsymbol{W}(k)]^T$$

and the system model matrices are

$$\boldsymbol{B} = -\boldsymbol{H}^T$$

and

$$\boldsymbol{C} = \boldsymbol{I}_{N\times N}$$

The system output is thus

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) = [\boldsymbol{p}^{\boldsymbol{obs}}(k)]^T$$

The control task can be accomplished by using the LQI control method [66] to design our controller. Fig. 6.1 shows the control block diagram for the system, in which $\boldsymbol{x}(k) \in \mathbb{R}^N$ is the system state, $\boldsymbol{y}(k) \in \mathbb{R}^N$ is the system output, $\boldsymbol{u}(k) \in \mathbb{R}^N$ is the controller output and $\boldsymbol{r}(k) \in \mathbb{R}^N$ is the controller input, which is the optimal collision probability $\boldsymbol{p}^*(k)$. $\boldsymbol{K} \in \mathbb{R}^{N\times 2N}$ is the control gain matrix, and $\boldsymbol{B} \in \mathbb{R}^{N\times N}$ and $\boldsymbol{C} \in \mathbb{R}^{N\times N}$ are the state-space system matrices. T_s is the sampling period of the system, i.e. the beacon interval 100ms.

The LQI controller computes an optimal state-feedback control law by minimising the quadratic cost function

$$J(\boldsymbol{u}(k)) = \sum_{k=0}^{\infty} \left(\boldsymbol{z}^T(k)\boldsymbol{Q}\boldsymbol{z}(k) + \boldsymbol{u}^T(k)\boldsymbol{R}\boldsymbol{u}(k)\right)$$

for any initial state $\boldsymbol{x}(0)$, in which $\boldsymbol{z}(k) = [\boldsymbol{x}(k); \boldsymbol{s}(k)]$ and $\boldsymbol{s}(k) = \boldsymbol{s}(k-1) + T_s \cdot$

$(\boldsymbol{r}(k-1) - \boldsymbol{y}(k-1))$ is the output of a discrete integrator.

The matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ are the weighting matrices respectively indicating the state and control cost penalties. $\boldsymbol{Q}$ and $\boldsymbol{R}$ are required to be real symmetric and positive definite.

The state feedback control law is defined as

$$\boldsymbol{u}(k) = -\boldsymbol{K}\boldsymbol{z}(k)$$

The optimal state feedback gain matrix $\boldsymbol{K}$ is computed by solving the associated discrete algebraic Riccati equation

$$\boldsymbol{P} = \hat{\boldsymbol{A}}^{\boldsymbol{T}}(\boldsymbol{P} - \boldsymbol{P}\hat{\boldsymbol{B}}(\boldsymbol{R} + \hat{\boldsymbol{B}}^{\boldsymbol{T}}\boldsymbol{P}\hat{\boldsymbol{B}})^{-1}\hat{\boldsymbol{B}}^{\boldsymbol{T}}\boldsymbol{P})\hat{\boldsymbol{A}} + \boldsymbol{Q}$$

in which

$$\hat{\boldsymbol{A}} = [\boldsymbol{0}_{N\times 2N};\quad -\boldsymbol{C} * T_s \quad \boldsymbol{I}_{N\times N}]$$

and

$$\hat{\boldsymbol{B}} = [\boldsymbol{B};\quad \boldsymbol{0}_{N\times N}]$$

$\boldsymbol{K}$ is constructed from the solution of the above algebraic Riccati equation $\boldsymbol{P^*}$ and weighting matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$, which is given by

$$\boldsymbol{K} = (\boldsymbol{R} + \hat{\boldsymbol{B}}^{\boldsymbol{T}}\boldsymbol{P}^*\hat{\boldsymbol{B}})^{-1}\hat{\boldsymbol{B}}^{\boldsymbol{T}}\boldsymbol{P}^*\hat{\boldsymbol{A}}$$

6.3.3 Selection of Q and R

The selection of weighting matrices Q and R affect the performance of the LQI controller. A simplified form using only 3 degrees of freedom is chosen in this work. the matrices are of the form

$$\boldsymbol{Q} = \begin{bmatrix} q_1\boldsymbol{I}_n & 0 \\ 0 & q_2\boldsymbol{I}_n \end{bmatrix}_{2N\times 2N} \tag{6.6}$$

and

$$\boldsymbol{R} = \rho \;.\; \boldsymbol{I}_{N\times N} \tag{6.7}$$

in which $q_1 \in \mathrm{R}^+$ is the weight for the state feedback cost; $q_2 \in \mathrm{R}^+$ is the weight for the integral feedback cost and $\rho \in \mathrm{R}^+$ is the weight for the input cost.

6.4 Simulation and Performance Evaluation

The main objective here is to achieve proportional fair allocation of station throughputs while satisfying specific delay constraints of different ACs. To verify that the

proposed algorithm meets this objective, we evaluate the throughput allocation and delay performance as done in previous chapters. The results are obtained using Matlab and Omnet++ discreet simulator based on the throughput analysis from equation 6.1 and the delay analysis Section 6.1. As the throughput and delay analysis is based on the 802.11e performance model presented in [36] under the assumptions that stations have saturated traffic and $CW_{max} = CW_{min}$. The accuracy of this network model has been fully verified in [36] for different network scenarios. It is therefore reasonable to use numerical results to verify the fairness optimisation algorithm proposed in this chapter.

We also note that in this chapter that we employed the state space representation of the wireless system, where the parameters of the state space system have been derived by converting the transfer function representation into the state space form using Matlab.

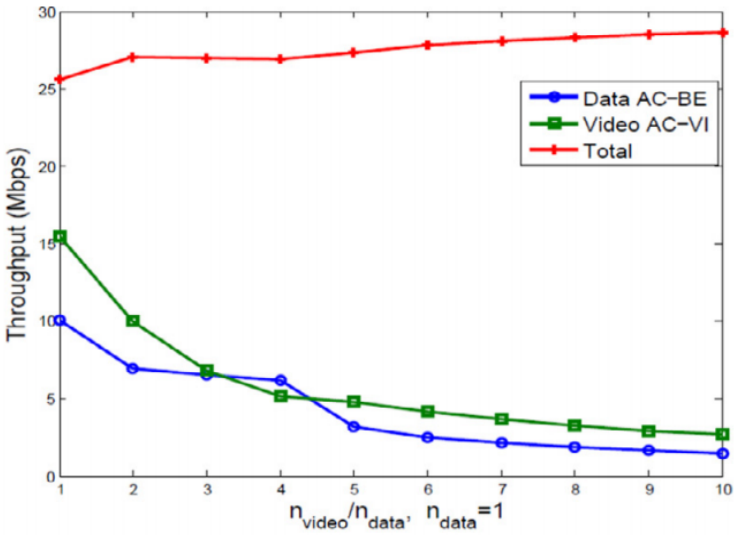

Figure 6.2: Throughput

An example of IEEE 802.11e WLAN with traffic of two ACs is considered, one of which is data traffic belonging to AC_{BE} (best effort), and the other is video traffic belonging to AC_{VI} (video). Flows of both ACs are saturated.

Fig.6.2 and Fig.6.3 shows the throughput and delay performance for two ACs versus the number of stations in AC_VI while keeping the number of stations in AC_BE fixed as 1. We let the TXOP burst reach the maximum limit as listed in Table 4.1. The average delay deadline for a single video packet is $d_2 = 250\mu$s, which is the successful transmission duration of a video packet. The delay deadline for a single data packet is $d_1 = 1000$ μs.

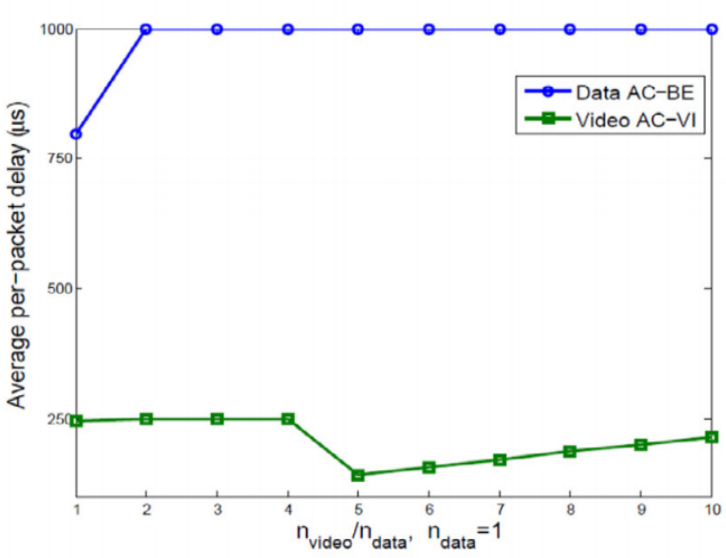

Figure 6.3: Delay

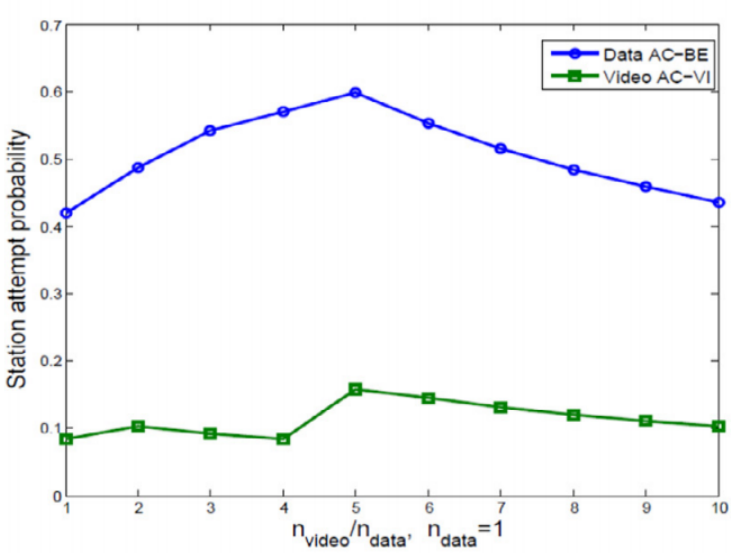

Figure 6.4: Station attempt probability

It can be seen that under these delay deadline constraints, the resource allocation can be divided into four phases. Phase I: When there are one video station and one data station, the network load is quite light, and both delay deadlines are not reached yet. Phase II: When the number of video stations increases to 2, the increased collision possibility leads to longer delays, and so the delay deadlines of both ACs are reached. As the number of video stations increases, in order to achieve a fair throughput allocation, data stations attempt more to access to the channel with an increased attempt probability, while video stations attempt less with a slightly decreasing attempt probability. The fairness algorithm makes the throughput of two ACs get closer to each other. Phase III: when the number of video stations increases up to 5, it comes to the turning point when video traffic is

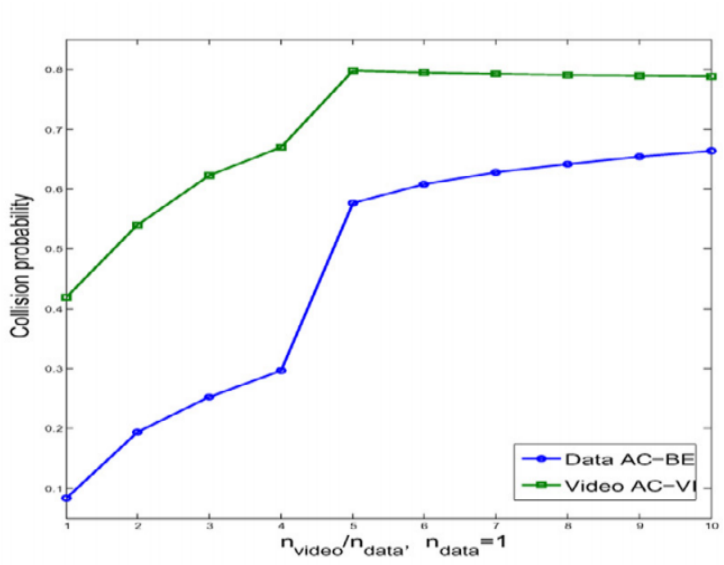

Figure 6.5: Collision probability

so aggressive that the proportional fairness algorithm allocates higher throughput to video traffic by assigning increased attempt probabilities to both ACs, but the increase for video traffic is larger than that for data traffic. The delay of video traffic is then reduced to be less than the deadline limit, while the delay constraint of data traffic remains tight. Phase IV: as the number of video stations continues increasing, the throughput ratio between video and data traffic remains around 1.5 in this phase. Even with 10 video stations and only one data station, the data station can still deliver a reasonable amount of throughput.

Fig.6.4 shows the corresponding station attempt probabilities. It can be seen that contrary to the 802.11e EDCA standard, our algorithm assigns data traffic with a higher attempt probability although it has lower priority. However not only does the delay performance satisfy the QoS requirement, the throughput is also fairly allocated between two ACs.

Air-time

The presence of collision losses and the coupling of station transmissions via carrier sense make the flow air-time in a WLAN not simply be the successful transmission duration but also include air-time as the fraction of time used for transmitting a flow, including both successful transmissions and collisions. For a flow of AC_i, the

flow total air-time is

$$T_i^{air} = \frac{P_i^{succ}T_i^{succ} + \tau_i P_i^{col}T^{col}}{P^eT^e + \sum_{i=0}^{N-1} P_i^{succ}T_i^{succ} + (1 - P^e - P^{succ})T^{col}}$$

$$= \frac{1}{X}\Big(\alpha_i\big(\frac{T_i^{succ}}{T^{col}} - 1\big) + \frac{\tau_i}{P^e}\Big)$$

The work in [17, 20] finds that the proportional fair allocation assigns equal total air-time to each flow in a WLAN, and the air-times sum to unity. We investigate the air-time allocation by considering an 802.11e WLAN with four ACs in which AC_{VI} and AC_{VO} have two stations, i.e. n_2=n_3=2 and AC_{BE} and AC_{BK} have one station, i.e. n_1=n_4=1. The PHY rates for the four ACs are the same i.e. r = 54Mbps. The packet sizeis L = 8000bits. Table 6.1 compares the flow total air-time allocations under different delay deadline constraints. In Case I, the average per-packet delay deadline for the four ACs are respectively d_1=900μs, d_2=300μs, d_3=250μs and d_4=1800μs, while in Case II, the average delay deadlines are relaxed as $d_1 = d_2 = d_3 = d_4$ = 5000μs.

Table 6.1: Comparison of flow total air-time allocation under different delay deadline constraints

Flow	1	2 and 3	4 and 5	6	Sum
AC	BE	VI	VO	BK	
Case I	0.1565	0.1530	0.1550	0.1562	0.9287
Case II	0.1667	0.1667	0.1667	0.1667	1

It can be seen that different from observations in [17, 20], the flow total air-time is not equalised in Case I, and the sum of air-times is less than 1. We also noticed that in this case, the delay constraints for voice traffic Flow 2 and 3 are tight. Nevertheless, when the delay constraints are relaxed in Case II, none of the delay deadline constraints is tight. Flows are now allocated with equal total air-times, and the air-times sum to 1. It is thus found that the air-time allocation

in our algorithm is affected by the imposed delay constraints. With tight delay constraints, the proportional fair allocation assigns flows with the exact amount of air-time needed by each of them. Air-time resource in the network is not completely occupied in such a case. However, with loose delay constraints, flows can occupy all the available air-time resources and air-time is evenly distributed amongst flows in a network as discovered in previous work. It is worth pointing out that since the flow air-time usage overlaps due to collisions, the flow total air-times summing to unity does not imply that the channel idle probability P^e=0

Q and R tuning

The tuning of q_1, q_2 and ρ can be performed using trial and error method. The influence of the three parameters are illustrated using an example with three ACs, AC_{BE}, AC_{VI}, AC_{VO}. The number of stations in AC_{BE}, AC_{VI} and AC_{VO} is respectively n_1=1, n_2=2 and n_3=1. Three ACs use the same PHY rate, i.e. $r_1 = r_2 = r_3$=54Mbps. The packet size is l=8000 bits. The average packet delay limit for AC_{BE}, AC_{VI} and AC_{VO} is respectively d_1=900μs, d_2 = 300μs and d_3 = 250μs. The TXOP burst reaches the maximum limit. Figures 6.4, 6.7 and 6.8 plots the system output response for AC_{BE} with different sets of q_1, q_2 and ρ values. The reason we do not put the output responses of AC_{VI} and AC_{VO} results is that we noticed that three outputs have the same convergence speed with a fixed set of q_1, q_2 and ρ values.

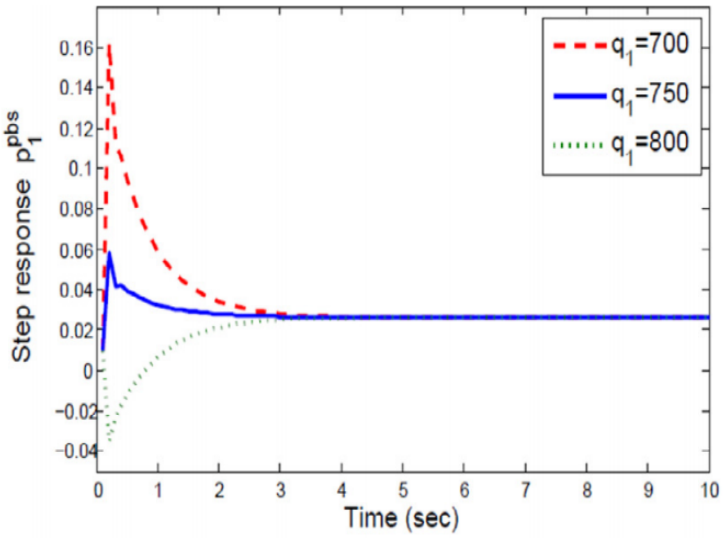

Figure 6.6: $q_2 = 2000, \rho = 0.005$

The effects of the three parameters are outlined as follows:

- q_1 imposes the constraints to the state dynamics. It is directly related to the overshoot. A higher q_1 corresponds to a lower overshoot. As shown in figure 6.6, q_1 =700 results in an overshoot while q_1 =800 corresponds to an undershoot.

- q_2 impacts on integral action dynamics and also on the system dynamics. As shown in Figure 6.7, the higher it is, the smaller rising time will be and the higher overshoot will be.

- ρ affects the dynamics of the controller input and also the system dynamics. It is related to the overshoot. Ahigher ρ results in a higher overshoot as shown in figure 6.8.

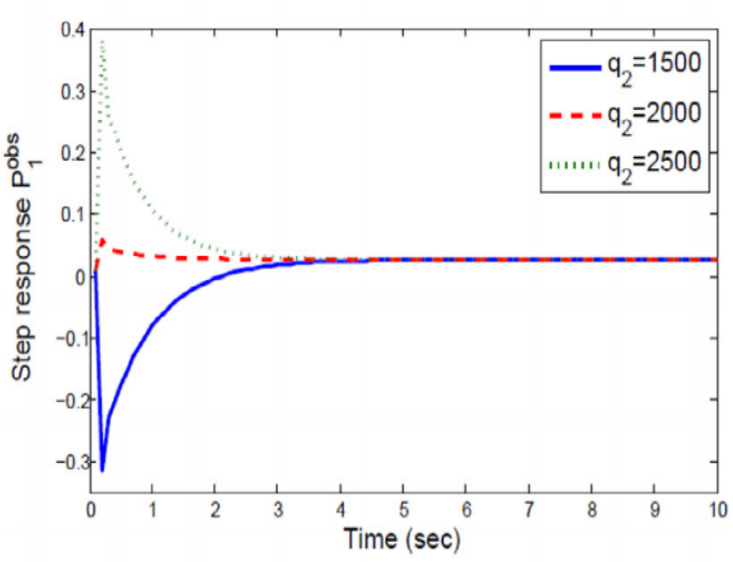

Figure 6.7: $q_1 = 750, \rho = 0.005$

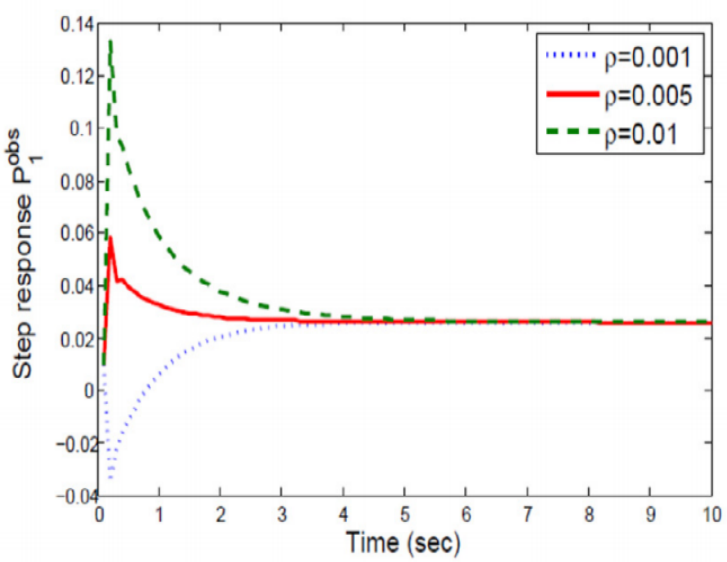

Figure 6.8: c $q_1 = 750, q_2 = 2000$

Controller adaptability to changes in the WLAN

We will next evaluate the adaptivity of the proposed method to the changes in the network size. The scenario being considered is depicted in figure 6.9. The algorithm starts at t=0s with four saturated stations, one in AC_{BE}, two in AC_{VI} and one in AC_{VO}. One more station in AC_{VO} joins the network at t=100s and leaves at t=200s. At t=300s one AC_{BK} station joins the network and after 100s one AC_{VI} leaves. The PHY data rates for the three ACs are the same, i.e. r=54Mbps. The packet size is l=8000 bits. the average packet delay limit for data, video, voice and background traffic are respectively d_1=900μs, d_2=300μs, d_3=250μs and d_4=1800μs. Figure 6.10 plots the variation of contention window over time. Figure 6.11 plots the corresponding station throughput for each AC. Q and R take the form as displayed in Eqs. 6.6 and 6.7. We choose $q_1 = 750$, q_2= 2000 and $\rho = 0.005$ to make a fast convergence speed. It can be seen that when the network condition changes the contention window converges to the desirable value very quickly as long as proper Q and R are chosen. Moreover, the steady-state errors

can be neglected, which means the control system has high accuracy performance.

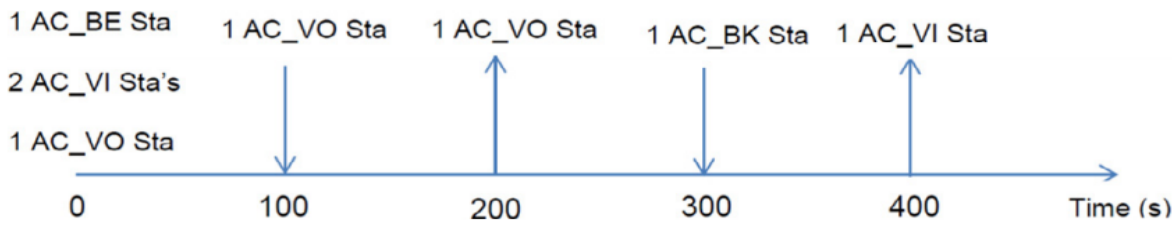

Figure 6.9: Injection and/or removal of stations in the WLAN

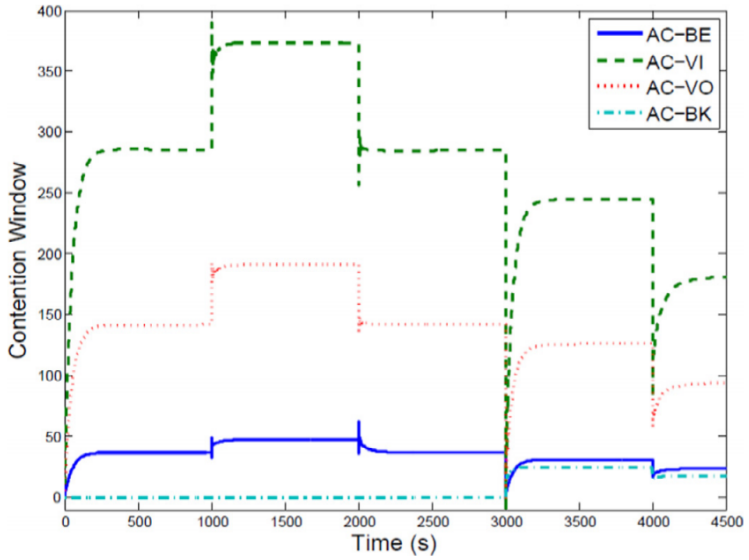

Figure 6.10: Contention Window over time

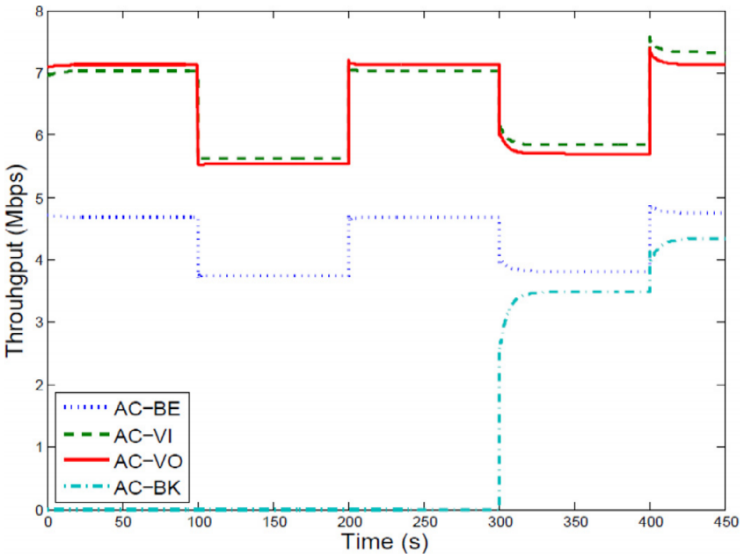

Figure 6.11: Station Throughput for each *AC*

Simulation results for Unsaturated Scenario

For the unsaturated scenario, we have maintained the same simulation parameters used for the saturated scenario above but assumed 50% degree of unsaturation i.e. $\rho = 0.5$. The results are as shown below:

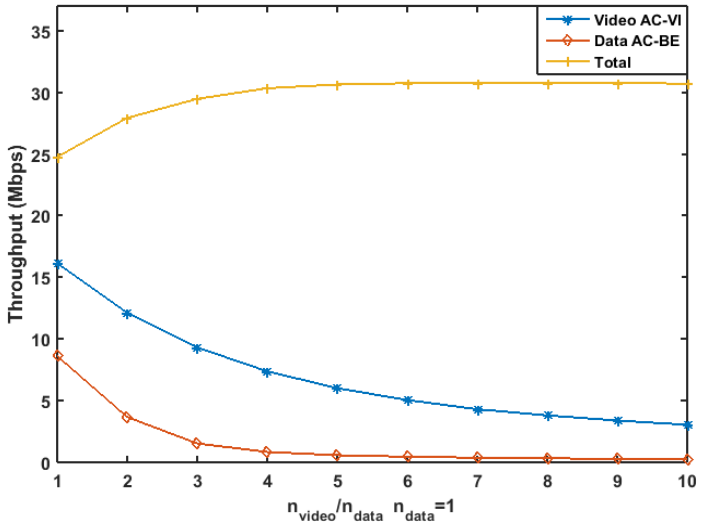

Figure 6.12: Station Throughput for Unsaturated Scenario

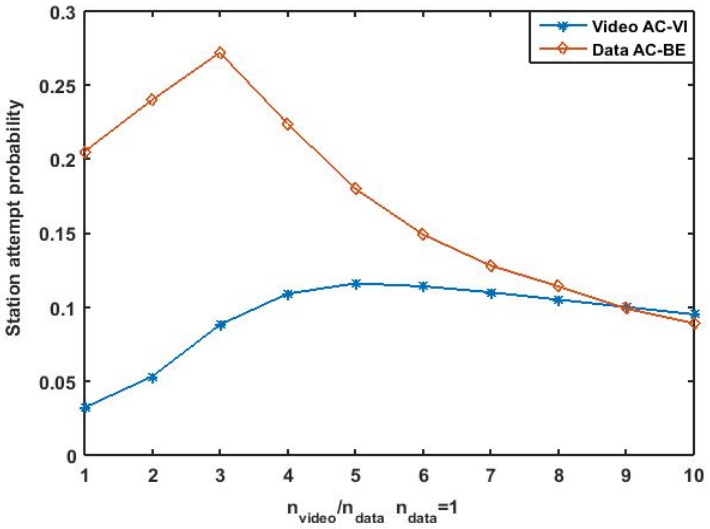

Figure 6.13: Station attempt probability for unsaturated scenario

Fig. 6.12, fig. 6.13 and fig 6.14 shows the throughput, station attempt probability and collision probability. We have maintained the same delay constraints used in section 6.4.

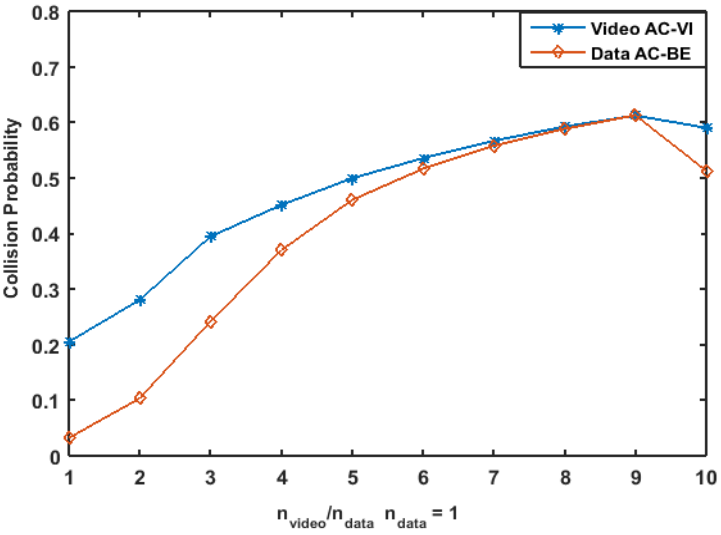

Figure 6.14: Collision probability for unsaturated scenario

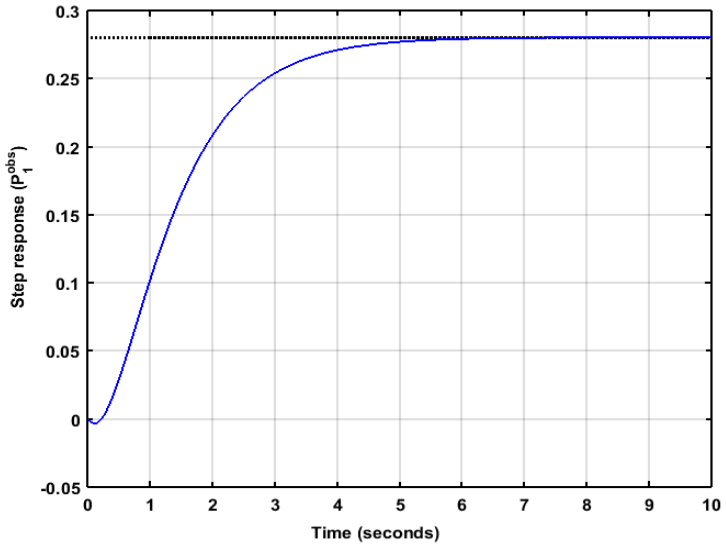

Figure 6.15: Step response for q_1=3000 q_2=200

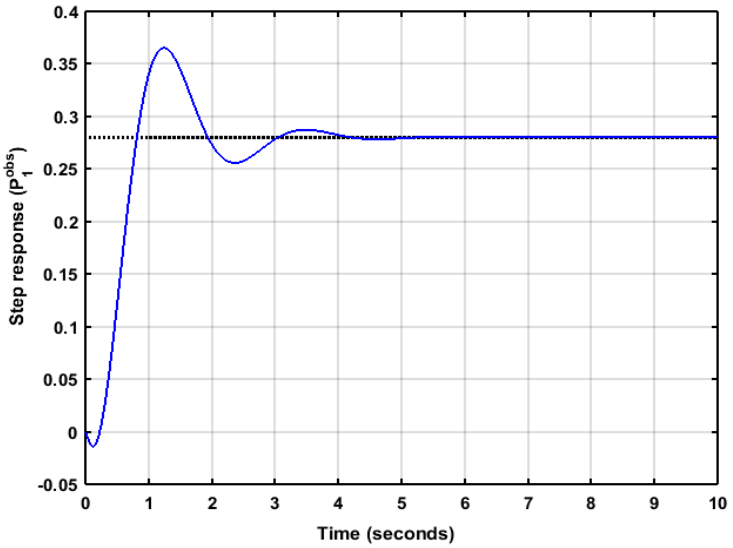

Figure 6.16: Step response for q_1=3000 q_2=200 with complex poles

Fig 6.15 shows the step response for q_1=3000 q_2=200. Fig 6.16 show the step response for the scenario where we attempted to tune the controller variable using the Control System Tool in Matlab. All these responses were still able to settle at the required reference point for the $n_{video}/n_{data} = 2$ scenario.

6.5 Conclusions

This chapter considers using a closed-loop control approach to achieve proportional fair allocation of station throughputs in a multi-priority EDCA WLAN. The optimal station attempt probability that leads to proportional fairness is derived given the average delay deadline constraints of different ACs present in an WLAN. To achieve the desirable proportional fairness, a centralised adaptive control approach is proposed. The WLAN is represented as a discrete MIMO LTI state-space model. The LQI control is used to tune the CW_{min} value to the optimum. We have demonstrated in simulations that the proposed control approach has high accuracy and fast convergence speed, and is adaptive to general network scenarios.

Chapter 7

Conclusions and Future Work

In this thesis, we have looked into the use of feedback control theory for the purpose of tuning the contention window value that is being broadcast by the access point to the nodes on the wireless network. The contention window value, used for contention purposes on a wireless network, have recommended values contained in the IEEE 802.11e standards and these are just recommended values only. We believe that the challenge of dynamically tuning this parameter can be solved by using the well established theory of feedback control and this thesis shows that the feedback control algorithm can be used on wireless networks without the need to adjust already existing protocol and hardware functionalities on the network.

In chapter 3 of this thesis, we developed a feedback control algorithm and we implemented this on a single-input-single-output network. We considered both throughput and probability of collision as the control variable. The result showed that the probability of collision gave contention window values that are closer to those recommended in the standards. We were also able to arrive at a controller with stable configurations and it can be tuned in order to achieve desired transient response.

In chapter 4, we proceeded to a multiple-input-multiple-output network with video and best effort traffic. Based on the result from chapter 3, we used the probability of collision as the controlled variable and in order to achieve fairness among contending traffic, we used the ratio of throughput and delay constraints as desired weights on the network. due to interferences experienced in MIMO networks, we introduced a decoupler to our controller algorithm. The results showed that the algorithm was able to give contention window outputs that maintained the desired throughput ration on the wireless network.

In chapter 5, we extended the work in the previous chapter by considering an

error-prone network while imposing a limit on the controller output. In this case, we set as limits the recommended values contained in the IEEE 802.11e standards. The results showed flexibility and versatility of the controller particularly for cases where the recommended values are to be maintained or where it is necessary to apply constraints on the contention window value.

In chapter 6, we then introduced the use of centralised adaptive control and we used the linear quadratic integral function to control the MIMO network.

In future, we hope to extend this work by using the Model Predictive Control (MPC) technique in achieving our aim of the outputting the optimum contention window value that enables the wireless network to function at its optimal state.

References

[1] Ibukunoluwa Akinyemi and Shuang-Hua Yang. Feedback control algorithm for optimal throughput in ieee 802.11 e edca networks. *Systems Science & Control Engineering*, 5(1):321–330, 2017.

[2] Brian D.O. Anderson and John B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice Hall International, 1990.

[3] A. Annese, G. Boggia, P. Camarda, LA Grieco, and S. Mascolo. Providing delay guarantees in IEEE 802.11e networks. In *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, volume 4, pages 2234–2238. IEEE, 2004.

[4] Abhinav Arora, Sung-Guk Yoon, Young-June Choi, and Saewoong Bahk. Adaptive TXOP allocation based on channel conditions and traffic requirements in IEEE 802.11 e networks. *IEEE Transactions on Vehicular Technology*, 59(3):1087–1099, 2010.

[5] Karl Johan Åström, Karl Henrik Johansson, and Q-G Wang. Design of decoupled PI controllers for two-by-two systems. *IEE Proceedings-Control Theory and Applications*, 149(1):74–81, 2002.

[6] Karl Johan Astrom and Richard M. Murray. *Feedback Systems*. Princeton University Press, 2008.

[7] K.J Astrom and B. Wittenmark. *Computer Controlled Systems: Theory and Design*. Prentice Hall, 3rd ed edition, 1997.

[8] Albert Banchs, Pablo Serrano, and Huw Oliver. Proportional fair throughput allocation in multirate IEEE 802.11 e wireless LANs. *Wireless Networks*, 13(5):649–662, 2007.

[9] Albert Banchs, Pablo Serrano, Paul Patras, and Marek Natkaniec. Providing throughput and fairness guarantees in virtualized WLANs through control theory. *Mobile Networks and Applications*, 17(4):435–446, 2012.

[10] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang. MACAW: a media access protocol for wireless LAN's. *ACM SIGCOMM Computer Communication Review*, 24(4):212–225, 1994.

[11] Giuseppe Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *Selected Areas in Communications, IEEE Journal on*, 18(3):535–547, 2000.

[12] Gennaro Boggia, Pietro Camarda, Luigi Alfredo Grieco, and Saverio Mascolo. Feedback-based bandwidth allocation with call admission control for providing delay guarantees in IEEE 802.11e networks. *Computer Communications*, 28(3):325–337, 2005.

[13] Gennaro Boggia, Pietro Camarda, Luigi Alfredo Grieco, and Saverio Mascolo. Feedback-based control for providing real-time services with the 802.11e MAC. *IEEE/ACM Transactions on Networking (TON)*, 15(2):323–333, 2007.

[14] Gennaro Boggia, Pietro Camarda, Claudio Di Zanni, Luigi A. Grieco, and Saverio Mascolo. *A dynamic bandwidth allocation algorithm for IEEE 802.11e WLANs with HCF access method*, pages 142–151. Quality for All. Springer, 2003.

[15] Dirceu Cavendish, Mario Gerla, and Saverio Mascolo. A control theoretical approach to congestion control in packet networks. *IEEE/ACM Transactions on Networking*, 12(5):893–906, 2004.

[16] Gabriele Cecchetti and Anna Lina Ruscelli. Performance evaluation of real-time schedulers for HCCA function in IEEE 802.11 e wireless networks. In *Proceedings of the 4th ACM symposium on QoS and security for wireless and mobile networks*, pages 1–8. ACM, 2008.

[17] Alessandro Checco and Douglas J Leith. Proportional fairness in 802.11 wireless LANs. *IEEE Communications Letters*, 15(8):807–809, 2011.

[18] WenTzu Chen. An effective medium contention method to improve the performance of IEEE 802.11. *Wireless Networks*, 14(6):769–776, 2008.

[19] Xiaomin Chen, Ibukunoluwa Akinyemi, and Shuang-Hua Yang. A control theoretic approach to achieve proportional fairness in 802.11 e EDCA WLANs. *arXiv preprint arXiv:1507.08920*, 2015.

[20] Xiaomin Chen and Douglas Leith. Proportional fair coding for 802.11 WLANs. *IEEE Wireless Communications Letters*, 1(5):468–471, 2012.

[21] Douglas Cooper. Proportional control, 2015.

[22] Joseph Dunn, Michael Neufeld, Anmol Sheth, Dirk Grunwald, and John Bennett. A practical cross-layer mechanism for fairness in 802.11 networks. In *Broadband Networks, 2004. BroadNets 2004. Proceedings. First International Conference on*, pages 355–364. IEEE, 2004.

[23] Juliana Freitag, Nelson LS Da Fonseca, and José F De Rezende. Tuning of 802.11 e network parameters. *IEEE Communications letters*, 10(8):611–613, 2006.

[24] Juan Garrido, Fernando Morilla, and Francisco Vazquez. Centralized PID control by decoupling of a boiler-turbine unit. In *Control Conference (ECC), 2009 European*, pages 4007–4012. IEEE, 2009.

[25] J. David Powell Gene F.Franklin and Abbas Emami-Naeini. *Feedback Control of Dynamic Systems*. Pearson Higher Education Inc., 6th ed edition, 2010.

[26] Luigi Fratta Giuseppe Bianchi and Matteo Oliveri. Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs. In *Personal, Indoor and Mobile Radio Communications, 1996. PIMRC'96., Seventh IEEE International Symposium on*, volume 2, pages 392–396. IEEE, 1996.

[27] Dawei Gong and Yuanyuan Yang. On-line AP association algorithms for 802.11 n WLANs with heterogeneous clients. *IEEE Transactions on Computers*, 63(11):2772–2786, 2014.

[28] LA Grieco, G. Boggia, S. Mascolo, and P. Camarda. A control theoretic approach for supporting quality of service in IEEE 802.11e WLANs with HCF. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 2, pages 1586–1591. IEEE, 2003.

[29] CV Hollot, Vishal Misra, Don Towsley, and W-B Gong. A control theoretic analysis of RED. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1510–1519. IEEE, 2001.

[30] Li Bin Jiang and Soung Chang Liew. Improving throughput and fairness by reducing exposed and hidden nodes in 802.11 networks. *IEEE Transactions on Mobile Computing*, 7(1):34–49, 2008.

[31] Sujay Parekh Joseph L. Hellerstein, Yixin Diao and Tilbury Dawn M. *Feedback Control of Computing Systems*. John Wiley sons, Inc, 2004.

[32] Tu Jun, Xia Xing, Chen Zhi-lan, Ye Zhi-wei, and Zhang Zhi. Improving throughput for heterogeneous traffic in IEEE 802. 11e EDCA. In *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2009. IDAACS 2009. IEEE International Workshop on*, pages 701–704. IEEE, 2009.

[33] Kenichi Kashibuchi, Abbas Jamalipour, and Nei Kato. Channel occupancy time based TCP rate control for improving fairness in IEEE 802.11 DCF. *IEEE Transactions on Vehicular Technology*, 59(6):2974–2985, 2010.

[34] Frank Kelly. Charging and rate control for elastic traffic. *European transactions on Telecommunications*, 8(1):33–37, 1997.

[35] Andrzej Kochut, Arunchandar Vasan, A Udaya Shankar, and Ashok Agrawala. Sniffing out the correct physical layer capture model in 802.11 b. In *Network Protocols, 2004. ICNP 2004. Proceedings of the 12th IEEE International Conference on*, pages 252–261. IEEE, 2004.

[36] Katarzyna KosekSzott, Marek Natkaniec, and Andrzej R. Pach. A simple but accurate throughput model for IEEE 802.11 EDCA in saturation and nonsaturation conditions. *Computer Networks*, 55(3):622–635, 2011.

[37] Jean-Yves Le Boudec. Rate adaptation, congestion control and fairness: A tutorial. *Web page, November*, 2005.

[38] Jeng Farn Lee, Wanjiun Liao, and Meng Chang Chen. Proportional fairness for QoS enhancement in IEEE 802.11 e WLANs. In *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on*, pages 1–pp. IEEE, 2005.

[39] Jeng Farn Lee, Wanjiun Liao, and Meng Chang Chen. A differentiated service model for enhanced distributed channel access (EDCA) of IEEE 802.11 e WLANs. *Mobile Networks and Applications*, 12(1):69–77, 2007.

[40] Lei Lei, Ting Zhang, Xiaoqin Song, Shengsuo Cai, Xiaoming Chen, and Jinhua Zhou. Achieving weighted fairness in WLAN mesh networks: An analytical model. *Ad Hoc Networks*, 25:117–129, 2015.

[41] Douglas J Leith, Qizhi Cao, and Vijay G Subramanian. Max-min fairness in 802.11 mesh networks. *IEEE/ACM Transactions on Networking (TON)*, 20(3):756–769, 2012.

[42] Douglas J Leith and Peter Clifford. TCP fairness in 802.11 e WLANs. In *Wireless Networks, Communications and Mobile Computing, 2005 International Conference on*, volume 1, pages 649–654. IEEE, 2005.

[43] Wei Li, Shengling Wang, Yong Cui, Xiuzhen Cheng, Ran Xin, Mznah A Al-Rodhaan, and Abdullah Al-Dhelaan. AP association for proportional fairness in multirate WLANs. *IEEE/ACM Transactions on Networking (TON)*, 22(1):191–202, 2014.

[44] Wan-Seon Lim and Young-Joo Suh. Achieving per-station fairness in IEEE 802.11 wireless LANs. In *World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a*, pages 1–9. IEEE, 2010.

[45] Cheng-Han Lin, Ce-Kuen Shieh, Wen-Shyang Hwang, and Wei-Tsang Huang. Proportional bandwidth allocation with consideration of delay constraint over IEEE 802.11 e-based wireless mesh networks. *Wireless Networks*, pages 1–18, 2016.

[46] Mohammad Malli, Qiang Ni, Thierry Turletti, and Chadi Barakat. Adaptive fair channel allocation for QoS enhancement in IEEE 802.11 wireless LANs. In *Communications, 2004 IEEE International Conference on*, volume 6, pages 3470–3475. IEEE, 2004.

[47] David Malone, Peter Clifford, and Douglas J Leith. MAC layer channel quality measurement in 802.11. *IEEE Communications Letters*, 11(2), 2007.

[48] Abdelhamid Nafaa, Adlen Ksentini, Ahmed Mehaoua, Y. Iraqi, and R. Boutaba. Sliding contention window (SCW): towards backoff range-based service differentiation over IEEE 802.11 wireless LAN networks. *Network, IEEE*, 19(4):45–51, 2005.

[49] Joe Naoum-Sawaya, Bissan Ghaddar, Sami Khawam, Haidar Safa, Hassan Artail, and Zaher Dawy. Adaptive approach for QoS support in IEEE 802.11e wireless LAN. In *Wireless And Mobile Computing, Networking And Communications, 2005.(WiMob'2005), IEEE International Conference on*, volume 2, pages 167–173. IEEE, 2005.

[50] Norman S. Nise. *Control Systems Engineering*. John Wiley sons, Inc, 6th ed edition, 2011.

[51] P.Albertos and A. Sala. *Multivariable Control Systems*. Springer, 2004.

[52] Eun-Chan Park, Nojun Kwak, Suk Kyu Lee, Jong-Kook Kim, and Hwangnam Kim. Provisioning QoS for wifi-enabled portable devices in home networks. *TIIS*, 5(4):720–740, 2011.

[53] Paul Patras, Albert Banchs, and Pablo Serrano. A control theoretic approach for throughput optimization in IEEE 802.11e EDCA WLANs. *Mobile Networks and Applications*, 14(6):697–708, 2009.

[54] Paul Patras, Albert Banchs, and Pablo Serrano. A control theoretic scheme for efficient video transmission over IEEE 802.11e EDCA WLANs. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 8(3):29, 2012.

[55] Paul Patras, Albert Banchs, Pablo Serrano, and Arturo Azcorra. A control-theoretic approach to distributed optimal configuration of 802.11 WLANs. *Mobile Computing, IEEE Transactions on*, 10(6):897–910, 2011.

[56] Riyadh Qashi, Martin Bogdan, and K. Hanssgen. Evaluating the QoS of WLANs for the IEEE 802.11 EDCF in real-time applications. In *Communications and Information Technology (ICCIT), 2011 International Conference on*, pages 32–35. IEEE, 2011.

[57] Lamia Romdhani, Qiang Ni, and Thierry Turletti. Adaptive EDCF: enhanced service differentiation for IEEE 802.11 wireless ad-hoc networks. In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, volume 2, pages 1373–1378. IEEE, 2003.

[58] Pablo Serrano, Albert Banchs, Paul Patras, and Arturo Azcorra. Optimal configuration of 802.11 e EDCA for real-time and data traffic. *Vehicular Technology, IEEE Transactions on*, 59(5):2511–2528, 2010.

[59] Pablo Serrano, Paul Patras, Andrea Mannocci, Vincenzo Mancuso, and Albert Banchs. Control theoretic optimization of 802.11 WLANs: implementation and experimental evaluation. *Computer Networks*, 57(1):258–272, 2013.

[60] Vasilios A Siris and George Stamatakis. Optimal cwmin selection for achieving proportional fairness in multi-rate 802.11 e WLANs: test-bed implementation and evaluation. In *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, pages 41–48. ACM, 2006.

[61] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. John Wiley sons, Inc, 2nd ed edition, 2005.

[62] IEEE Standard. Part 11: Wireless lan medium access control (MAC) and physical layer (PHY) specifications ieee standard for information technology, telecommunications and information exchange between systems;. 2012.

[63] Chiapin Wang. Achieving per-flow and weighted fairness for uplink and downlink in IEEE 802.11 WLANs. *EURASIP Journal on Wireless Communications and Networking*, 2012(1):239, 2012.

[64] Chien-Erh Weng and Ho-Lung Hung. Performance analysis of priority schemes for IEEE 802.11 e wireless local area networks using multiple flows distributed weighted fair queuing algorithm. *Universal Journal of Communications and Network*, 2(1):14–21, 2014.

[65] Xin-Wei Yao, Wan-Liang Wang, Shuang-Hua Yang, and Yue-Feng Cen. Bio-inspired self-adaptive rate control for multi-priority data transmission over wlans. *Computer Communications*, 53:73–83, 2014.

[66] Peter Colin Young and JC Willems. An approach to the linear multivariable servomechanism problem. *International journal of control*, 15(5):961–979, 1972.

[67] Zhenhui Yuan and Gabriel-Miro Muntean. A prioritized adaptive scheme for multimedia services over IEEE 802.11 WLAN s. *IEEE Transactions on Network and Service Management*, 10(4):340–355, 2013.