**UNIVERSITY CARLOS III OF MADRID**

**Department of Telematics Engineering**


Master of Science Thesis

# A Control Theoretic Approach for Throughput Optimisation in IEEE 802.11e EDCA WLANs


Author: **Paul Horaţiu Pătraş**
Dipl.Eng. in Telecommunications

Supervisors: **Albert Banchs Roca, Ph.D.**
**Pablo Serrano Yáñez-Mingot, Ph.D.**


Leganés, July 2008

# Abstract

The MAC layer of the 802.11 standard, based on the CSMA/CA mechanism, specifies a set of parameters to control the aggressiveness of stations when trying to access the channel. However, these parameters are statically set independently of the conditions of the WLAN (e.g. the number of contending stations), leading to poor performance for most scenarios. To overcome this limitation previous work proposes to adapt the value of one of those parameters, namely the CW, based on an estimation of the conditions of the WLAN. However, these approaches suffer from two major drawbacks: $i$) they require extending the capabilities of standard devices or $ii$) are based on heuristics.

This thesis proposes a control theoretic approach to adapt the CW to the conditions of the WLAN, based on an analytical model of its operation, that is fully compliant with the 802.11e standard. We use a Proportional Integrator controller in order to drive the WLAN to its optimal point of operation and perform a theoretic analysis to determine its configuration. We show by means of an exhaustive performance evaluation that our algorithm maximises the total throughput of the WLAN and substantially outperforms previous standard-compliant proposals.

# Table of Contents

# List of Figures

# Chapter 1

# Introduction

Wireless Local Area Networks (WLANs) have become a popular solution for providing Internet connectivity anywhere and anytime due to their low cost and ease of deployment. The IEEE 802.11 standard [18] defines two mechanisms for accessing the medium: a centralised one, named the point coordination function (PCF) and a distributed one, the distributed coordination function (DCF).

For the basic access mechanism, the standard defines a set of parameters that controls the way stations access the channel. Since radio channels have limited resources, they are shared by multiple users and multimedia applications need to be supported, optimising throughput and providing fair treatment to all stations require optimal configuration of the channel access parameters. In particular, the Contention Window (CW) parameter controls the probability that a station defers or transmits a frame once the medium has become idle, and its value has a strong impact on the throughput of the WLAN.

The CW configuration used by the 802.11 standard [18] is statically set, independently of the number of contending stations. This static configuration leads to poor performance in most scenarios. In particular, when there are many stations in the WLAN it would be desirable to have larger CWs in order to avoid too frequent collisions, while with few stations smaller CWs would help to reduce the channel idle time. Following this, it has been shown that for a given number of actively contending stations there exists an optimal CW configuration that maximises throughput performance [7, 4].

Following the above observations, many authors have proposed to dynamically adapt the CW by estimating the number of active stations in the WLAN. These works can be classified in two different groups:

1. Distributed approaches [8, 25, 22, 23, 24, 3, 14, 10, 9, 26], that require every node on the WLAN to implement a mechanism for adjusting the backoff behaviour. The main disadvantage of these approaches is that they change the rules of the 802.11 standard and require introducing modifications to the existing hardware.

2. Centralised approaches [21, 13], based on a single node that periodically distributes the set of MAC layer parameters to be used by every station. These approaches are compatible with the 802.11e standard. However, because they are based on heuristic algorithms and lack analytical support, they do not guarantee optimal performance.

In this thesis we propose a novel adaptive algorithm to dynamically adjust the CW configuration of 802.11-based Wireless LANs. We share the same goal with previous approaches, i.e., to maximise the overall throughput performance of the wireless network. Compared to the existing schemes our proposal benefits from the following key improvements:

- It is fully compatible with the 802.11e standard and does not require any modification to existing hardware, since the dynamic adjustment is based only on observing successfully received frames at the Access Point (AP).

- It is based on a well established scheme from discrete-time control theory, namely the Proportional Integrator (PI). We optimally tune the parameters of the PI controller by conducting a control theoretic analysis of the system.

The rest of the thesis is structured as follows. In Chapter 2 we briefly describe the EDCA mechanism of the IEEE 802.11e standard and present different existing CW control proposals. In Chapter 3 we analyse the throughput performance of EDCA and find the configuration for which this is optimised. Next we present the proposed algorithm, which aims at driving the system to the optimal operation by using a PI controller. The parameters of the controller are set following a control theoretical analysis of our system. The performance of the proposed scheme is validated by means of simulation experiments in Chapter 4. Finally, Chapter 5 provides a summary of the thesis and identifies future work.

# Chapter 2

# State of the Art

In this chapter we first present the EDCA mechanism of IEEE 802.11a which supports service differentiation as well as reconfiguration of the MAC parameters. Next we summarise the principles of several previous proposals that aim to adjust the MAC parameters of the WLANs with the goal of optimising the throughput. Nevertheless, we list a set of rules that have been taken into account when designing our CW control algorithm.

## 2.1 IEEE 802.11e EDCA

This section briefly summarises the EDCA mechanism. This mechanism has been defined in the 802.11e standard [19] and will be included in the ongoing new revision of the 802.11 standard [20].

DCF is the fundamental channel access scheme in IEEE 802.11 WLANs, based on the carrier sense multiple access with collision avoidance (CSMA/CA) mechanism [7]. Two ways of accessing the channel are defined. The default mechanism assumes a 2-way handshake, i.e. after each transmission an acknowledgement frame is sent to confirm the correct reception of the frame (basic access mechanism). The second method is optional and requires a 4-way handshaking procedure, known as RTS/CTS. A station will try to reserve the channel prior to a transmission by sending a request-to-send frame (RTS). The receiving station will reply with a clear-to-send (CTS) frame if no collision was involved and the basic transmission-acknowledge mechanism will begin. This aims to minimize the time a station spends resolving collisions but also to avoid the hidden-terminal problem [11].

EDCA (Enhanced Distributed Channel Access) regulates the access to the wireless channel on the basis of the *channel access functions* (CAFs). A station may run up to 4 CAFs, and each of the frames generated by the station is mapped to one of them. Once a station becomes active, each CAF executes an independent backoff process to transmit its frames.

A station with a new frame to transmit monitors the channel activity. If the medium is idle for a period of time equal to the arbitration interframe space parameter ($AIFS$), the CAF transmits. Otherwise, if the channel is sensed busy (either immediately or during the $AIFS$ period), the CAF continues to monitor the channel until it is measured idle for an $AIFS$ time, and, at this point, the backoff process starts.

Upon starting the backoff process, a random value uniformly distributed in the range $(0, CW-1)$ is chosen and the backoff time counter is initialized with this number. The $CW$

Table 2.1: IEEE 802.11b MAC parameters

| **Parameter** | **Value** |
|---|---|
| Backoff slot time | 20 $\mu$s |
| DIFS | 50 $\mu$s |
| SIFS | 10 $\mu$s |
| $CW_{min}$ | 32 |
| $CW_{max}$ | 1024 |

value is called the contention window, and depends on the number of failed transmissions of a frame. At the first transmission attempt, $CW$ is set equal to the minimum contention window parameter ($CW_{min}$).

As long as the channel is sensed idle, the backoff time counter is decremented once every empty slot time $T_e$. 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. This period is equal to $AIFS$ if the transmission is received with a correct FCS, and $EIFS - DIFS + AIFS$ otherwise, where $EIFS$ (the extended interframe space) and $DIFS$ (the distributed interframe space) are physical layer constants.

As soon as the backoff time counter reaches zero, the CAF transmits its frame. A collision occurs when two or more CAFs start transmitting simultaneously. An acknowledgement (Ack) frame is used to notify the transmitting station that the frame has been successfully received. The Ack is sent upon the reception of the frame, after a period of time equal to the physical layer constant SIFS (the short interframe space).

If the Ack is not received within a time interval given by the $Ack\_Timeout$ physical layer constant, the CAF assumes that the frame was not received successfully. The transmission is then rescheduled by re-entering the backoff process, which starts at an $AIFS$ time following the timeout expiry. After each unsuccessful transmission, $CW$ is doubled, up to a maximum value given by the $CW_{max}$ parameter. If the number of failed attempts reaches a predetermined retry limit $R$, the frame is discarded. This mechanism is illustrated in Fig. 2.1 for a simple scenario with two active stations using a minimum AIFS parameter (equal to DIFS) and the default IEEE 802.11b MAC parameters. The values of these parameters are listed in Table 2.1.

After a (successful or unsuccessful) frame transmission, before sending the next frame, the CAF must execute a new backoff process. As an exception to this rule, the protocol allows the continuation of an EDCA transmission opportunity (TXOP). A continuation of an EDCA TXOP occurs when a CAF retains the right to access the channel following the completion of a transmission. In this situation, the station is allowed to send a new frame a SIFS period after the completion of the previous one. The period of time a CAF is allowed to retain the right to access the channel is limited by the transmission opportunity limit parameter ($TXOP\_limit$).

In the case of a single station running more than one channel access function, if the backoff time counters of two or more CAFs reach zero at the same time, a scheduler inside the station avoids the *internal collision* by granting the access to the channel to the highest

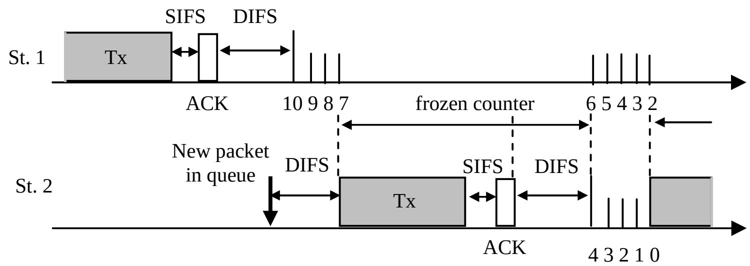


Figure 2.1: Example of EDCA mechanism.

priority CAF. The other CAFs of the station involved in the internal collision react as if there had been a collision on the channel, doubling their $CW$ and restarting the backoff process.

As it can be seen from the description of EDCA given in this section, the behaviour of a CAF depends on a number of parameters, namely $CW_{min}$, $CW_{max}$, $AIFS$ and $TXOP\_limit$. These are configurable parameters that can be set to different values for different CAFs. The CAFs are grouped by Access Categories (ACs), all the CAFs of an AC having the same configuration. The Access Point (AP) announces periodically (every 100 ms) the parameters of each AC by means of beacon frames.

Following the above considerations several adaptive algorithms have been proposed trying to adjust the value of the contention window, such that throughput optimisation is achieved. The works published until now are mainly focusing on two approaches. One consists of observing the channel state and modifying the $CW$ at every station (distributed method), e.g. [8, 25, 22, 23, 24, 3, 14, 10, 9, 26] and the second assumes monitoring the contention level at the access point, tuning the $CW$ appropriately and distributing it to the associated stations (centralised method), e.g. [13, 21]. With the approval of the IEEE 802.11e standard [19], changing of the $CW$ value is now allowed to be performed by means of beacon frames with an adjustment granularity equal to the beacon interval (100ms).

## 2.2 Distributed CW Adjustment Mechanisms

The other approach to adaptive $CW$ control involves monitoring of the network state at every station and independently adjusting the configuration of the EDCA parameters. The advantages of such techniques are the following:

- They do not require estimation of the number of station actively contending for the channel
- Each station's adjustment mechanism is independent from the one employed by other stations.

### 2.2.1 Slow Decrease Algorithm

The default 802.11 DCF adapts its $CW$ to the current congestion level by doubling its value upon a collision and by resetting it upon a successful transmission. The work in

[22] considers that a successful transmission doesn't reflect a congestion level decrease, but rather a convenient $CW$ value. Therefore, the $CW$ value should be maintained as long as the congestion level remains the same. The second assumption made by this proposal is that the congestion level is not likely to decrease sharply. Hence, by resetting $CW$ to $CW_{min}$ the risk of experiencing collisions and retransmissions until reaching a high value again is large. This also leads to channel underutilisation. The slow decrease (SD) algorithm attempts to minimize the number of collisions during congestion, but, by keeping the same $CW$ values when congestion sharply drops, increases the overhead and may decrease the throughput. The slow decrease factor $\delta$ is in the range (0,1). The behaviour of the SD algorithm is shown below.

**Algorithm 1** Slow Decrease Algorithm

**if** $Tx\_successful$ **then**
  $CW_{new} \leftarrow max(CW_{min}, \delta \cdot CW_{old})$
**else**
  $CW_{new} \leftarrow min(2 \cdot CW_{old}, CW_{max})$
**end if**

### 2.2.2 CW Idle-Slots-based Control

Another distributed $CW$ adjustment scheme was proposed by Xia et al. in [25]. The $CW$ Idle-Slots-based Control (WISC) is significantly different from the standard BEB mechanism. The adjustment scheme is based on a control-theoretic approach that dynamically adjusts the $CW$ based on the locally available channel state. The average number of consecutive idle slots between two transmissions is monitored and then used to drive the $CW$ to its optimal value, such that throughput is maximised. This mechanism is extremely accurate but requires large processing overhead, since the $CW$ is adjusted after each transmitted frame.

**Algorithm 2** CW Idle-Slots-based Control Algorithm

**while** Tx attempt **do**
  $e_{prev} \leftarrow e_{cur}$
  $e_{cur} \leftarrow I_m - I(t)$
  $CW_{cur} \leftarrow CW_{cur} + C_1 \cdot e_{cur} + C_2 \cdot e_{prev}$
  **if** $CW_{cur} < CW_{min}$ **then**
    $CW_{cur} \leftarrow CW_{min}$
  **end if**
  **if** $CW_{cur} > CW_{max}$ **then**
    $CW_{cur} \leftarrow CW_{max}$
  **end if**
**end while**

According to the demonstration in [7] the throughput of the WLAN is maximized for an optimal value of the $CW$. Following this observation, the authors represent the optimal channel state $I_m$ as the optimal average number of idle slots between two transmissions. This also holds the optimal $CW$ value. Hence, by tuning the $CW$ such that the number of idle slots $I(t)$ is around the optimal $I_m$ value, the throughput of the network will be optimized. The optimal value of idle-slots between two transmissions has been previously estimated by [15] to be $I_m \approx 5.68$. The formal description of WISC is illustrated in Algorithm 2. The values of the $C_1$ and $C_2$ coefficients have been derived using analytical methods form control theory and are 11.75 and 5.75 respectively.

Although outperforming the default DCF mechanism, distributed solutions suffer from some significant drawbacks in terms of their practical use. First of all, they require modifications of the station's hardware and drivers since the backoff rules are embedded inside them. Moreover, each station requires additional CPU time for running the algorithm locally and additional transmission delay is introduced. In what follows we will present a set of different approaches that try to overcome these limitations by adjusting the WLAN parameters in a single node and distributing them to the other stations.

## 2.3 Centralised CW Adjustment Mechanisms

In the IEEE 802.11e standard [19] it is specified that the WLAN configuration can be changed by the AP using beacons. The values of $CW_{min}$ and $CW_{max}$ can be distributed to all the stations every beacon interval. Several algorithms making use of this feature have been proposed to dynamically adjust the value of the $CW$ such that throughput is optimized. In what follows we will present the mechanisms employed by two such solutions, namely the Sliding Contention Window (SCW) [21] and the algorithm presented in [13], hereafter referred to as Dynamic Tuning Algorithm (DTA). The advantages of these approaches are the following:

- They do not require any modification of the hardware or drivers on the stations
- They react to changes in the network load

### 2.3.1 Sliding Contention Window

SCW was designed to maximise network utilisation and provide traffic differentiation by modifying the ranges from which the backoff counter is chosen. The sliding contention window of each traffic class has a lower bound $CW^{LB}$ and an upper bound $CW^{UB}$. These bounds change but remain within the interval $(CW_{min}, CW_{max})$. To adjust the $CW$ range, SCW uses a linear-increase linear-decrease model, increasing or decreasing the SCW range in steps of a sliding factor SF. When a flow experiences high loss rate, the SCW is increased by a step of SF until it reaches the upper bound, while if the loss rate is low, instead of resetting the contention window value, the range is decreased until the lower bound reaches $CW_{min}$. The adjustment of the $CW$ range for best effort traffic is based on the observation of the instantaneous network load $B(T)$. $B(T)$ is the fraction of slots the medium was detected busy out of the previous $T$ slots. If the network load drops below a threshold $B_{Th}$, the SCW range is decreased, while if it exceeds the throughput saturation threshold $B_{Sat}$ the range will be increased. The authors have chosen by means of simulation the values for $B_{Th}$ and $B_{Sat}$, 0.7 and 0.9 respectively [21]. The algorithm is presented below.

**Algorithm 3** Sliding Window Algorithm

**if** $B(T)/T \leq B_{Th}$ **then** ▷ Decrease
  **if** $oldCW^{LB} - SF \geq CW_{min}$ **then**
    $newCW^{LB} \leftarrow oldCW^{LB} - SF$
    $newCW^{UB} \leftarrow oldCW^{UB} - SF$
  **else**
    $newCW^{LB} \leftarrow CW_{min}$
    $newCW^{UB} \leftarrow CW_{min} + size(SCW)$
  **end if**
**else if** $B_{Th} < B(T)/T < B_{Sat}$ **then**
  maintain $SCW$
**else if** $B(T)/T \geq B_{Sat}$ **then** ▷ Increase
  **if** $oldCW^{UB} + SF \leq CW_{max}$ **then**
    $newCW^{LB} \leftarrow oldCW^{LB} + SF$
    $newCW^{UB} \leftarrow oldCW^{UB} + SF$
  **else**
    $newCW^{LB} \leftarrow CW_{max} - size(SCW)$
    $newCW^{UB} \leftarrow CW_{max}$
  **end if**
**end if**

Table 2.2: Parameters of the SCW algorithm

| **Parameter** | **Value (time slots)** |
|---|---|
| $SCW_{size}$ | 256 |
| $SF$ | 128 |
| $CW_{min}$ | 128 |
| $CW_{max}$ | 1024 |

The parameters used by the authors for adjusting the $CW$ range under best effort traffic conditions are shown in Table 2.2.

### 2.3.2 Dynamic Tuning Algorithm

J. Freitag et al. proposed a different centralised algorithm that is based on estimating the number of stations served by an AP [13]. Their assumption is that when there are only few stations, a low $CW_{min}$ will reduce the station idle time and increase the channel utilization. Also, when the number of stations is high, $CW_{min}$ should be increased to avoid collisions. The number of active stations may be determined by the AP by estimating the number of active flows and identifying their source address. Following this reasoning the dynamic tuning algorithm (DTA) will double the value of $CW_{min}$ if the number of active stations is larger than $CW_{min}$. Otherwise, if $CW_{min}/2$ is greater than the number of stations, it will be reduced to half of its value. $CW_{min}$ is kept the same if the number of stations is between

$CW_{min}$ and $CW_{min}/2$. Moreover, the algorithm checks whether the values of $CW_{min}$ are below a minimum value. In this case, they will be set to the minimum value allowed. The AP will inform the stations of the new value of $CW_{min}$ using beacons.

**Algorithm 4** Dynamic Tuning Algorithm

```
if num_stations > CW_min then
    CW_min ← CW_min · 2 + 1
end if
if num_stations < CW_min/2 then
    CW_min ← (CW_min − 1)/2
end if
if CW_min < CWMinimum then
    CW_min ← CWMinimum
end if
```

## 2.4 Optimal Configuration Prerequisites

Our goal is to find the EDCA parameters that maximise the throughput of the WLAN, while fairly sharing the bandwidth among the competing stations. Following this goal, we use the following configuration for the stations:

- In an optimally configured WLAN collisions are very infrequent. Hence, the impact of avoiding some of them due to multiple CAFs is negligible. Consequently, we will assume each station executes a single CAF and transmits one frame upon accessing the channel.

- As proved in [6], if using an $AIFS$ parameter different from the minimum value ($DIFS$), there exists at least another configuration with minimum $AIFS$ that provides equal or better throughput performance. Consequenty, we set the $AIFS$ parameter to the minimum value ($DIFS$) for all stations.

- All stations will contend with the same $CW_{min}$ and $CW_{max}$ parameters in order to guarantee equal probability of accessing the channel and same throughput guarantees.

In what follows we design an adaptive algorithm that adjusts the configuration of $CW_{min}$ and $CW_{max}$ with the goal of maximising the overall WLAN throughput. This algorithm is executed at the AP, which uses beacon frames to announce the computed $CW_{min}$ and $CW_{max}$ values to the stations.

# Chapter 3

# A Control Theoretic Approach for Throughput Optimisation

In this chapter we conduct a mathematical analysis of the WLAN throughput and then we derive the conditions for which this is maximised. Then we present the principles of our CW configuration algorithm and analyse our control system from a theoretical viewpoint. Finally, we show how we configure our controller to achieve a proper trade-off between stability and speed of reaction to changes.

## 3.1 Throughput Analysis and Optimisation

In this chapter we present a throughput analysis of an EDCA WLAN configured according to the rules listed in section 2.4. Based on this analysis, we find the collision probability of an optimally configured WLAN, which is the basis of the algorithm presented in the following section.

We start by analysing the case when all stations are saturated[1] and consider later the case when some stations are not saturated. Let us define $\tau$ as the probability that a saturated station transmits at a randomly chosen slot time. This can be computed according to [7] as follows:

$$\tau = \frac{2}{1 + W + pW \sum_{i=0}^{m-1} (2p)^i} \tag{3.1}$$

where $W = CW_{min}$, $m$ is the maximum backoff stage ($CW_{max} = 2^m CW_{min}$) and $p$ is the probability that a transmission collides. In a WLAN with $n$ stations,

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

The throughput obtained by a station can be computed as follows

$$r = \frac{P_s l}{P_s T_s + P_c T_c + P_e T_e} \tag{3.3}$$

where $l$ is the packet length, $P_s$, $P_c$ and $P_e$ are the probabilities of a success, a collision and an empty slot time, respectively, and $T_s$, $T_c$ and $T_e$ are the respective slot time durations.

[1]Following [7], by saturation we mean that a station always has a packet ready for transmission.

The probabilities $P_s$, $P_c$ and $P_e$ are computed as

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

$$P_e = (1-\tau)^n \tag{3.5}$$

$$P_c = 1 - n\tau(1-\tau)^{n-1} - (1-\tau)^n \tag{3.6}$$

and the slot time durations $T_s$ and $T_c$ as

$$T_s = T_{PLCP} + \frac{H}{C} + \frac{l}{C} + SIFS + T_{PLCP} + \frac{Ack}{C} + DIFS \tag{3.7}$$

$$T_c = T_{PLCP} + \frac{H}{C} + \frac{l}{C} + DIFS \tag{3.8}$$

where $T_{PLCP}$ is the PLCP (Physical Layer Convergence Protocol) preamble and header transmission time, $H$ is the MAC overhead (header and FCS), Ack is the size of the acknowledgement frame and $C$ is the channel bit rate.

The above terminates our throughput analysis. We next address, based on this analysis, the issue of optimizing the throughput performance of the WLAN. To this aim, we can rearrange Eq. (3.3) to obtain

$$r = \frac{l}{T_s - T_c + \frac{P_e(T_e - T_c) + T_c}{P_s}} \tag{3.9}$$

As $l$, $T_s$, and $T_c$ are constants, maximizing the following expression will result in the maximization of $r$,

$$\hat{r} = \frac{P_s}{P_e(T_e - T_c) + T_c} \tag{3.10}$$

Given $\tau \ll 1$, $\hat{r}$ can be approximated by

$$\hat{r} = \frac{n\tau - n(n-1)\tau^2}{T_e - n(T_e - Tc)\tau + \frac{n(n-1)}{2}(T_e - T_c)\tau^2} \tag{3.11}$$

The optimal value of $\tau$, $\tau_{opt}$, that maximizes $\hat{r}$ can then be obtained by

$$\left.\frac{d\,\hat{r}}{d\,\tau}\right|_{\tau=\tau_{opt}} = 0 \tag{3.12}$$

which neglecting the terms of higher order than 2 yields

$$a\tau^2 + b\tau + c = 0 \tag{3.13}$$

where

$$a = -\frac{n^2(n-1)}{2}(T_c - T_e) \tag{3.14}$$

$$b = -2n(n-1)T_e \tag{3.15}$$

$$c = nT_e \tag{3.16}$$

Isolating $\tau_{opt}$ from the above yields

$$\tau_{opt} = \sqrt{\left(\frac{2T_e}{n(T_c - T_e)}\right)^2 + \frac{2T_e}{n(n-1)(T_c - T_e)}} - \frac{2T_e}{n(T_c - T_e)} \tag{3.17}$$

Given $T_e \ll T_c$, we finally obtain the following approximate solution for the optimal $\tau$,

$$\tau_{opt} \approx \frac{1}{n}\sqrt{\frac{2T_e}{T_c}} \tag{3.18}$$

With the above $\tau_{opt}$, the corresponding optimal collision probability is equal to

$$p_{opt} = 1 - (1 - \tau_{opt})^{n-1} = 1 - \left(1 - \frac{1}{n}\sqrt{\frac{2T_e}{T_c}}\right)^{n-1} \tag{3.19}$$

which can be approximated by

$$p_{opt} \approx 1 - e^{-\sqrt{\frac{2T_e}{T_c}}} \tag{3.20}$$

This implies that, under optimal operation with saturated stations, the collision probability in the WLAN is a constant independent of the number of stations. The key approximation of this paper is to assume that, when some of the stations are saturated and some are not, the optimal collision of the WLAN takes the same constant value.

In the following section we design an adaptive algorithm that adjusts the WLAN configuration with the goal of driving the collision probability to the above value. Note that, since this a constant value, our algorithm does not need to know the number of stations in the WLAN.

## 3.2 Adaptive Algorithm

We next present our adaptive algorithm; this algorithm runs at the AP and consists of the following two steps which are executed iteratively:

- During the period between two beacon frames (which lasts 100 ms), the AP measures the collision probability of the WLAN resulting from the current CW configuration.
- At the end of this period, the AP computes the new CW configuration based on the measured collision probability and distributes it to the stations in a new beacon frame.

Our algorithm uses a PI controller to drive the WLAN to its optimal point of operation. In the following, we explain how the CW configuration is adjusted using a control signal. We then describe our system from a control theoretical standpoint. Next, we analyse our system by linearising the behaviour of the WLAN. Finally, we use this analysis to adequately configure the parameters of the PI controller.

### 3.2.1 CW Configuration

Following the previous section, our goal is to adjust the $CW$ parameters of EDCA ($CW_{min}$ and $CW_{max}$) in order to force the collision probability given by Eq. (3.20). Since the default $CW$ values given by the 802.11e standard[2] ($CW_{min}^{default}$ and $CW_{max}^{default}$) are typically too small, yielding a too aggressive behaviour, in order to achieve optimal operation these $CW$ parameters should be increased.

Following the above reasoning, our algorithm increases the default $CW_{min}$ of the standard by some $CW_{offset}$,

$$CW_{min} = CW_{min}^{default} + CW_{offset} \tag{3.21}$$

while keeping the default value for the maximum backoff stage, i.e.

$$CW_{max} = 2^m CW_{min} \tag{3.22}$$

where $m$ is the maximum backoff stage of the default configuration.

In order to ensure that our algorithm never underperforms the standard default configuration by using overly small $CW$ values, we force that $CW_{offset}$ cannot take negative values, which guarantees that $CW_{min}$ will never take smaller values than the standard's default. In addition, we also force that $CW_{offset}$ cannot take values that yield a $CW_{min}$ larger than $CW_{max}^{default}$. These bounds provide a safeguard against too large and too small values of $CW_{min}$, respectively. In the rest of the paper we assume that $CW_{offset}$ always takes values within these bounds and do not further consider this effect.

### 3.2.2 Control System

From a control theoretic standpoint, our system can be seen as the composition of the two modules depicted in Figure 3.1: the *controller* C(z), which is the adaptive algorithm that controls the WLAN, and the *controlled system* H(z), which is the WLAN itself. In our proposal we use for the controller module a classical scheme from discrete-time control theory, namely the *Proportional Integrator* (PI) *Controller*.

Following the above, our control system consists of the following two modules:

- The controller module located at the AP, that is based on the *Proportional Integrator* (PI) controller. The AP estimates the collision probability and provides it to the controller, which takes as input the difference between the estimated collision probability and its desired value as given by Eq. (3.20). With this input, the controller computes the $CW_{offset}$ value.

- The controlled module is the 802.11e EDCA WLAN system. As specified by the standard, the AP distributes the new $CW$ configuration to the stations every 100 ms. This configuration is obtained from the $CW_{offset}$ value given by the controller, following Eqs. (3.21) and (3.22).

[2]Although the 802.11e parameters are configurable, the standard includes a default setting for these parameters [19].

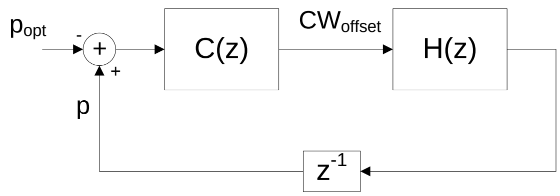


Figure 3.1: Control system.

The estimation of the collision probability over a 100 ms period is performed at the AP as follows. Let $S$ be the number of frames received by the AP during this period with the retry bit unset, and $R$ be the number of frames received with the retry bit set. Then, if we assume that no frames are discarded due to reaching the retry limit, the collision probability $p$ can be computed as

$$p = \frac{R}{R+S} \tag{3.23}$$

since the above is precisely the probability that the first transmission attempt of a frame collides.

Note that with the above method, the AP can compute the probability $p$ by simply analysing the header of the frames successfully received, which can be easily done with no modifications to the AP's hardware and driver.

### Transfer Function Characterisation

In order to analyse our system from a control theoretic standpoint, we need to characterise the Wireless LAN system with a transfer function that takes $CW_{offset}$ as input and gives the collision probability $p$ as output. Since the collision probability is measured every 100 ms interval, we can safely assume that the obtained measurement corresponds to stationary conditions and therefore the system does not have any memory. With this assumption,

$$p = 1 - (1-\tau)^{n-1} \tag{3.24}$$

where $\tau$ is a function of $CW_{offset}$ as given by Eq. (3.1),

$$\tau = \frac{2}{1 + (CW_{min}^{default} + CW_{offset})(1 + p\sum_{i=0}^{m-1}(2p)^i)} \tag{3.25}$$

The above equations give a nonlinear relationship between $p$ and $CW_{offset}$. In order to express this relationship as a transfer function, we linearise this relationship when the system is perturbed around its stable point of operation[3], i.e.

$$CW_{offset} = CW_{offset,opt} + \delta CW_{offset} \tag{3.26}$$

where $CW_{offset,opt}$ is the $CW_{offset}$ value that yields the optimal collision probability $p_{opt}$ computed in Eq. (3.20).

[3] A similar approach was used in [16] to analyze RED from a control theoretical standpoint.

With the above, the oscillations of the collision probability around its point of operation $p_{opt}$ can be approximated by

$$p \approx p_{opt} + \frac{\partial p}{\partial CW_{offset}} \delta CW_{offset} \tag{3.27}$$

The above partial derivative can be computed as

$$\frac{\partial p}{\partial CW_{offset}} = \frac{\partial p}{\partial \tau} \frac{\partial \tau}{\partial CW_{offset}} \tag{3.28}$$

where

$$\frac{\partial p}{\partial \tau} \approx n - 1 \tag{3.29}$$

and

$$\frac{\partial \tau}{\partial CW_{offset}} = -\frac{2(1 + p\sum_{i=0}^{m-1}(2p)^i)}{\left(1 + CW_{min}(1 + p\sum_{i=0}^{m-1}(2p)^i)\right)^2} \tag{3.30}$$

Evaluating the partial derivative at the stable point of operation $p = p_{opt}$, and using the approximation $p_{opt} \approx (n-1)\tau_{opt}$ given by Eq. (3.19) and the expression for $\tau_{opt}$ given by Eq. (3.1), yields

$$\frac{\partial p}{\partial CW_{offset}} \approx -p_{opt}\tau_{opt} \frac{1 + p_{opt}\sum_{i=0}^{m-1}(2p_{opt})^i}{2} \tag{3.31}$$

If we now consider the transfer function that allows us to characterize the perturbations of $p$ around its stable point of operation as a function of the perturbations in $CW_{offset}$,

$$\delta P(z) = H(z)\, \delta CW_{offset}(z) \tag{3.32}$$

we obtain from Eqs. (3.27) and (3.31) the following expression for the transfer function,

$$H(z) = -p_{opt}\tau_{opt} \frac{1 + p_{opt}\sum_{i=0}^{m-1}(2p_{opt})^i}{2} \tag{3.33}$$

Figure 3.2 illustrates the above linearised model when working around its stable operation point:

$$\begin{cases} p = p_{opt} + \delta p \\ CW_{offset} = CW_{offset,opt} + \delta CW_{offset} \end{cases} \tag{3.34}$$

Note that, as compared to the model of Figure 3.1, in Figure 3.2 only the perturbations around the stable operation point are considered.

### Controller Configuration

We next address the issue of configuring the PI controller.

The transfer function of the controller is given by

$$C(z) = K_p + \frac{K_i}{z - 1} \tag{3.35}$$

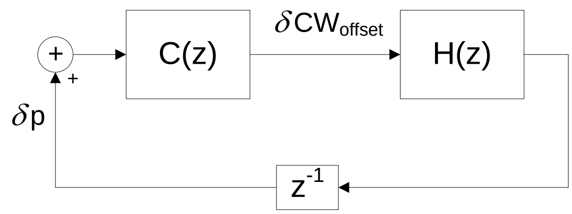


Figure 3.2: Linearised system.

We observe from the above transfer function that the PI controller depends on the following two parameters to be configured: $K_p$ and $K_i$. Our goal in the configuration of these parameters is to find the right trade-off between speed of reaction to changes and stability. To this aim, we use the Ziegler-Nichols rules [12] which have been designed for this purpose. These rules are applied as follows. First, we compute the parameter $K_u$, defined as the $K_p$ value that leads to instability when $K_i = 0$, and the parameter $T_i$, defined as the oscillation period under these conditions. Then, $K_p$ and $K_i$ are configured as follows:

$$K_p = 0.4K_u \tag{3.36}$$

and

$$K_i = \frac{K_p}{0.85T_i} \tag{3.37}$$

In order to compute $K_u$ we proceed as follows. The system is stable as long as the absolute value of the closed-loop gain is smaller than 1,

$$|H(z)C(z)| = K_p p_{opt} \tau_{opt} \frac{1 + p_{opt} \sum_{i=0}^{m-1} (2p_{opt})^i}{2} < 1 \tag{3.38}$$

which yields the following upper bound for $K_p$,

$$K_p < \frac{2}{p_{opt}\tau_{opt}(1 + p_{opt} \sum_{i=0}^{m-1} (2p_{opt})^i)} \tag{3.39}$$

Since the above is a function of $n$ (note that $\tau_{opt}$ depends on $n$) and we want to find an upper bound that is independent of $n$, we proceed as follows. From Eq. (3.19), we observe that $\tau_{opt}$ is never larger than $p_{opt}$ for $n > 1$ (note that for $n = 1$ the system is stable for any $K_p$). With this observation, we obtain the following constant upper bound (independent of $n$):

$$K_p < \frac{2}{p_{opt}^2(1 + p_{opt} \sum_{i=0}^{m-1} (2p_{opt})^i)} \tag{3.40}$$

Following the above, we take $K_u$ as the value where the system may turn unstable (given by the previous equation),

$$K_u = \frac{2}{p_{opt}^2(1 + p_{opt} \sum_{i=0}^{m-1} (2p_{opt})^i)} \tag{3.41}$$

and set $K_p$ according to Eq. (3.36),

$$K_p = \frac{0.4 \cdot 2}{p_{opt}^2(1 + p_{opt}\sum_{i=0}^{m-1}(2p_{opt})^i)} \tag{3.42}$$

With the $K_p$ value that makes the system become unstable we have $H(z)C(z) = -1$. With such a closed-loop transfer function, a given input value changes its sign at every time slot, yielding an oscillation period of two slots ($T_i = 2$). Thus, from Eq. (3.37),

$$K_i = \frac{0.4}{0.85p_{opt}^2(1 + p_{opt}\sum_{i=0}^{m-1}(2p_{opt})^i)} \tag{3.43}$$

which completes the configuration of the PI controller. The stability of this configuration is guaranteed by Theorem 1, included in the Appendix.

# Chapter 4

# Performance Evaluation

In order to evaluate the performance of the proposed algorithm, we performed an exhaustive set of simulation experiments. For this purpose, we have extended the simulator used in [6, 5]; this is an event-driven simulator that closely follows the details of the MAC protocol of 802.11 EDCA. For all tests, we used a payload size of 1000 bytes and the system parameters of the IEEE 802.11b physical layer [17]. For the simulation results, average and 95% confidence interval values are given (note that in many cases confidence intervals are too small to be appreciated in the graphs). Unless otherwise stated, we assume that all stations are saturated.

## 4.1 Throughput Performance

The main objective of the proposed algorithm is to maximize the throughput performance of the WLAN. To verify if the proposed algorithm meets this objective, we evaluated the total throughput obtained for different numbers of stations $n$. As a benchmark against which to assess the performance of our approach, we compared it against the static optimal configuration given by Eq. (3.18) and the default configuration given in the 802.11e standard [19]. Note that the static optimal configuration method requires the knowledge of the number of active stations, which challenges its practical use.

The results of the experiment described above are given in Figure 4.1. We can observe from the figure that the performance of the proposed algorithm follows very closely the static optimal configuration in terms of total throughput. In contrast, the default configuration performs well for a small number of stations but sees its performance substantially degraded as the number of stations increases. From these results, we conclude that the proposed algorithm maximises the throughput performance.

## 4.2 Stability

One of the objectives of the configuration of the PI controller presented in Section 3.2.2 is guaranteeing a stable behaviour of the system. In order to assess this objective, we plot in Figure 4.2 the value of the system's control signal ($CW_{offset}$) every beacon interval, for our $\{K_p, K_i\}$ setting with $n = 20$ stations. We can observe that with the proposed setting,

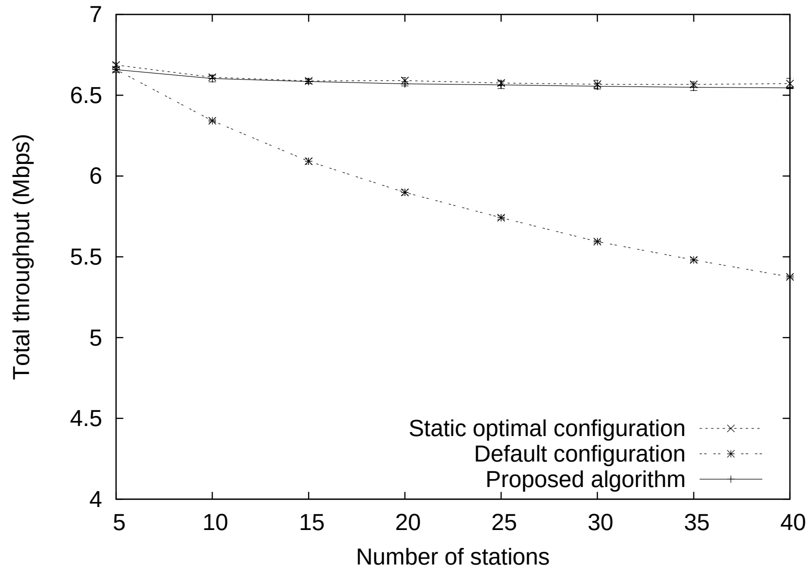


Figure 4.1: Throughput performance.

$CW_{offset}$ performs stably with minor deviations around its point of operation. In case that a larger setting for $\{K_p, K_i\}$ was used to improve the speed of reaction to changes, we would have the situation of Figure 4.3. For this case, with values for $\{K_p, K_i\}$ 20 times larger, the $CW_{offset}$ shows a strong unstable behaviour with drastic oscillations. We conclude that the proposed configuration achieves the objective of guaranteeing a stable behaviour.

## 4.3 Speed of Reaction to Changes

In addition to a stable behaviour, we also require the PI controller to quickly react to changes on the WLAN. To assess this objective we ran the following experiment. For a WLAN with 15 saturated stations, at $t = 80$ we added 15 more stations. We plot the behaviour of $CW_{offset}$ for our $\{K_p, K_i\}$ setting in Figure 4.4 (label "$Kp, Ki$"). The system reacts fast to the changes on the WLAN, as $CW_{offset}$ reaches the new value almost immediately. We have already shown in the previous section that large values for the parameters of the controller lead to unstable behavior. To analyze the impact of small values for these parameters, we plot on the same figure the $CW_{offset}$ evolution for a $\{K_p, K_i\}$ setting 20 times smaller (label "$Kp/20, Ki/20$"). With such setting, although obtaining a minor gain in stability, the system reacts too slow to changes of the conditions on the WLAN.

We conclude that, by means of the Ziegler-Nichols rules, we achieve a proper trade-off between stability and speed of reaction to changes. To further validate this, in Figure 4.5 we illustrate the time plot of the instantaneous throughput of one station, averaged over 1 second intervals, for the same previous experiment of Figure 4.4. We can see from the figure that the system is able to provide stations with constant throughput (apart from minor oscillations due to the use of CSMA/CA), reacting almost immediately to changes.

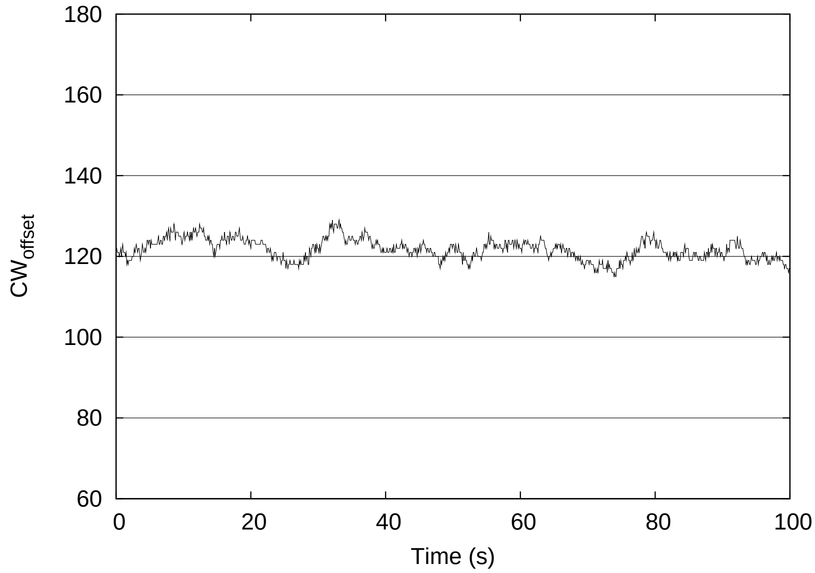


Figure 4.2: Stable configuration.

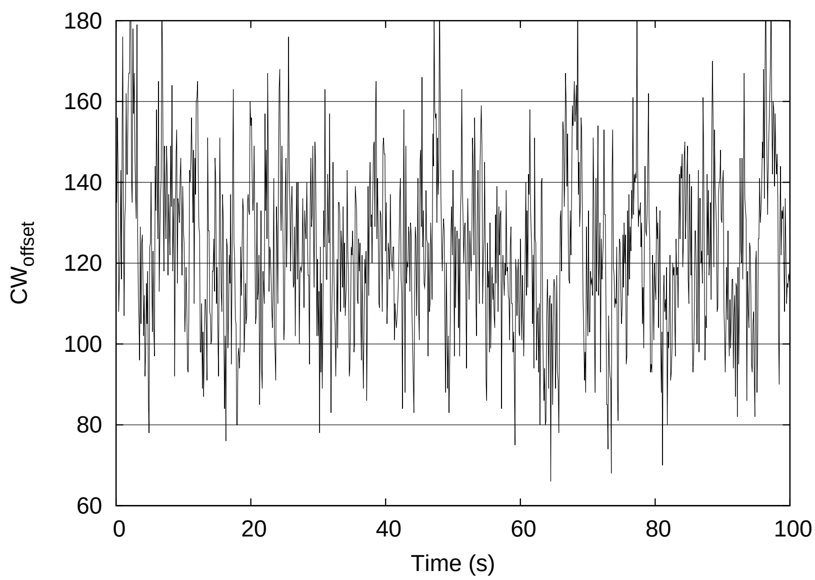


Figure 4.3: Unstable configuration.

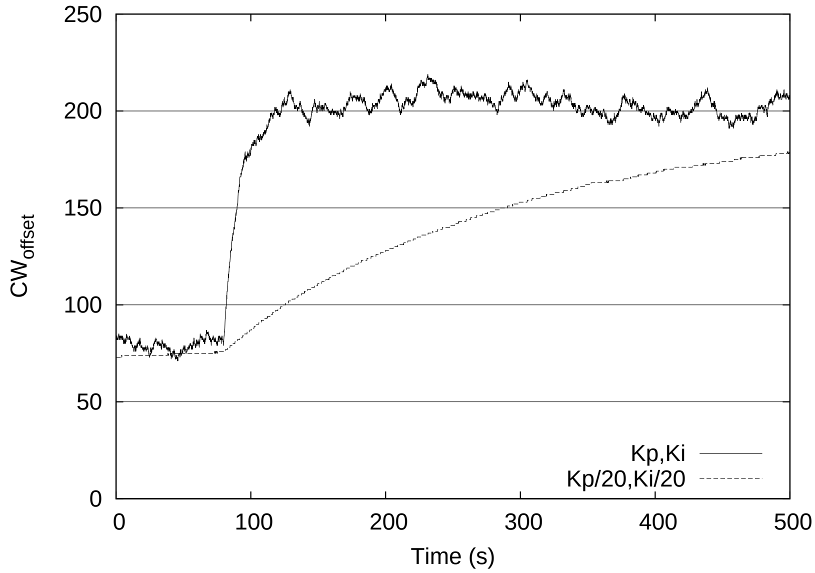


Figure 4.4: Speed of reaction to changes.

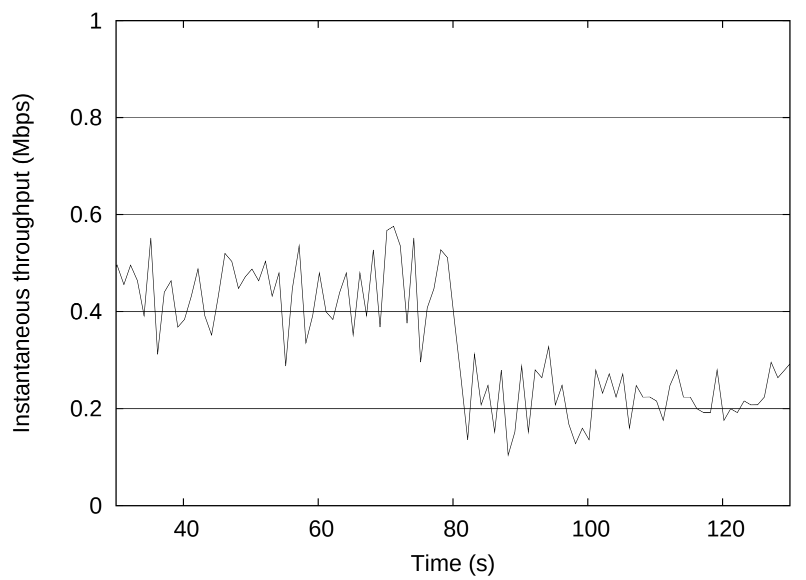


Figure 4.5: Instantaneous throughput.

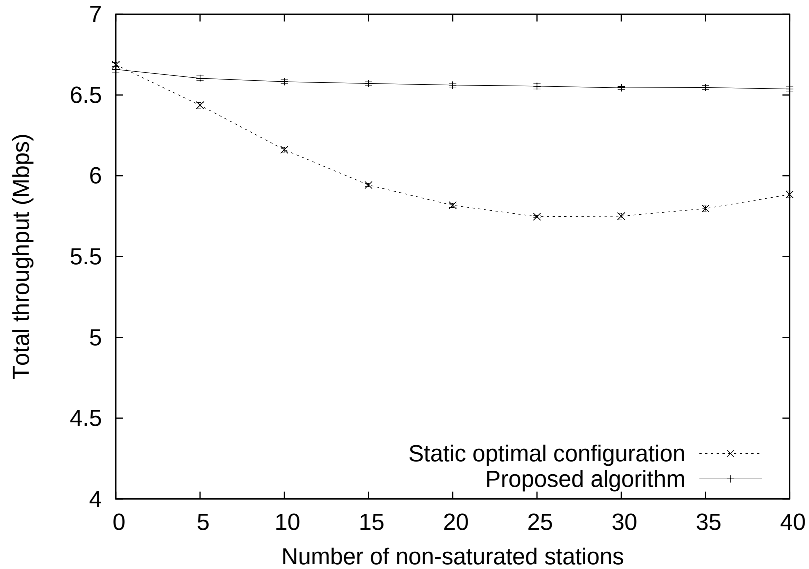


Figure 4.6: Non-saturated stations.

## 4.4 Non-saturated Stations

Our approach has been designed to optimise performance both under saturation and non-saturation conditions, in contrast to the static optimal configuration shown previously which is based on the assumption that all stations are saturated. In order to evaluate and compare the performance of the two algorithms when there are non-saturated stations in addition to saturated stations, we performed the following experiment. We had 5 saturated stations and a variable number of non-saturated stations in the WLAN. The non-saturated stations generated CBR traffic at rate of 100 Kbps. The total throughput resulting from this experiment is illustrated in Figure 4.6. In this figure, we compare the performance of our approach against the static optimal configuration, resulting from computing the configuration with Eq. (3.18) and taking as $n$ the total number of stations present in the WLAN, regardless of whether they are saturated or not.

We observe from Figure 4.6 that with our approach, the total throughput remains approximately constant with values similar to the ones obtained for saturation conditions (Figure 4.1), independently of the number of non-saturated stations. In contrast, the performance of the static optimal configuration decreases very substantially as the number of non-saturated stations increases. This is due to the fact that the static optimal configuration considers that all stations are continuously sending packets and therefore uses too conservative CW values.

From the above results we conclude that our algorithm achieves optimal performance also when non-saturated stations are present in the WLAN, in contrast to the static optimal configuration which sees its performance severely degraded as the number of non-saturated stations increases.

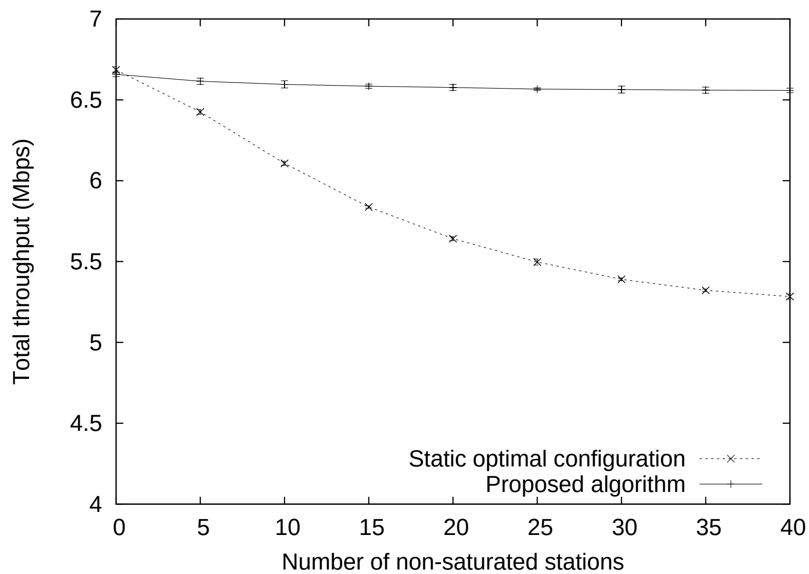


Figure 4.7: Bursty traffic.

## 4.5 Bursty Traffic

In order to understand whether bursty traffic can harm the performance of the proposed algorithm, we repeated the experiment reported in the previous section but with the non-saturated stations sending highly bursty traffic instead of CBR. In particular, in our experiment we used ON/OFF sources with exponentially distributed active and idle periods of an average duration of 100 ms each. The results of this experiment are depicted in Figure 4.7.

We can see from these results that, similarly to Figure 4.6, the proposed algorithm performs optimally independent of the number of bursty stations, and substantially outperforms the static optimal configuration. We conclude that our approach does not only work well under constant traffic but also under highly variable sources.

## 4.6 Comparison Against Other Approaches

The Sliding Contention Window (SCW) [21] and the dynamic tuning algorithm (DTA) of [13] are, like ours, centralised solutions compatible with the 802.11e standard that do not require hardware modifications. In this section we compare our solution against these centralised mechanisms.

Figure 4.8 gives the total throughput performance of the different solutions for varying numbers of stations. We observe that the proposed algorithm outperforms significantly both SCW and DTA. The reason is that our algorithm is sustained on the analysis of Section 3.1, which guarantees optimised performance, in contrast to SCW and DTA which are based on heuristics. In particular, SCW uses an algorithm to adjust $CW_{min}$ that chooses overly large values, thereby degrading the performance. On the other hand, DTA sets the $CW_{min}$ value as an heuristic function of the number of stations yielding overly small values, which results in a degraded performance also for this case.

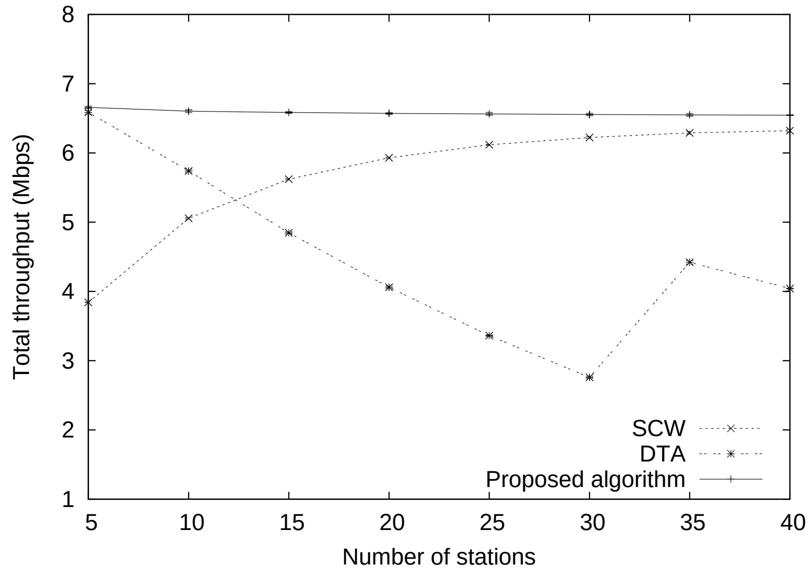


Figure 4.8: Comparison against other approaches.

## 4.7 Non-ideal Channel

In all previous simulations we have considered an ideal channel. The experimental study conducted in [2] showed that one of the non-ideal effects that we have in a real channel is the so called *capture effect*. This effect occurs when, upon a collision in the channel, the frame received with the strongest signal survives the collision and is *captured* by the receiving station[1].

In order to evaluate the impact of the capture effect on our algorithm, we performed an experiment in which a collision involving several stations was captured by the receiver with a given probability. The station whose frame was captured was randomly chosen among the ones involved in the collision. Figure 4.9 shows the result of this experiment for $n = 20$ stations and different capture probabilities in the range $\{0, 1\}$.

The results obtained confirm that our algorithm works well in a non-ideal channel and outperforms both the standard configuration (for small capture probabilities) and the static optimal configuration (for large ones).

[1] Another non-ideal effect consists of the *channel errors*. However, these occur infrequently in current multi-rate Wireless LANs such as 802.11a/b/g. With these technologies, channel errors trigger link adaptation mechanisms to use modulation schemes more robust to errors.

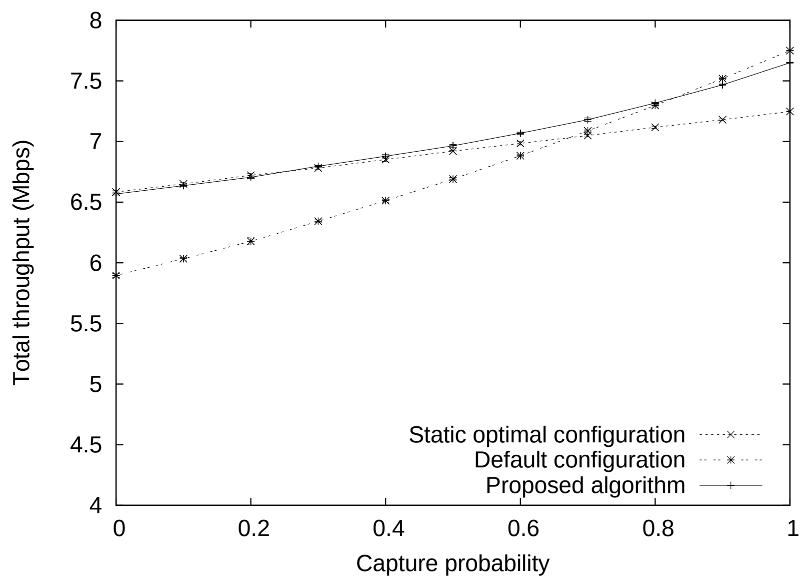


Figure 4.9: Capture effect.

# Chapter 5

# Summary

In this thesis we have proposed a novel adaptive algorithm for optimising the throughput performance of WLANs. The algorithm is sustained on the observation that the collision probability in an optimally configured WLAN is approximately constant, independent of the number of stations. Our proposal only requires to measure this collision probability by monitoring successfully transmitted frames during an inter-beacon period at the AP.

Our algorithm is based on a well established controller from discrete-time control theory, the PI controller. By means of a theoretical analysis of the WLAN and the controller, we have designed our algorithm to maximise the throughput performance. We achieve a proper trade-off between stability and speed of reaction to changes by applying the Ziegler-Nichols rules. We have shown via simulations that our algorithm drives the WLAN to the optimal point of operation, even for non-saturated and highly bursty traffic, reacting quickly to changes of the conditions in the WLAN.

As opposed to most of the previous proposals, or algorithm is fully compatible with the 802.11e EDCA standard and does not require any modifications neither at a hardware nor at a driver level. We have shown that our proposal substantially outperforms other centralised 802.11e-compatible solutions.

Our future work will focus on evaluating the performance of the proposed algorithm by means of practical simulations. The impact of non-ideal channel effects such as errors, fading and interference will be analysed in a real environment. Also, the throughput performance of our mechanisms will be validated under mixed traffic conditions, considering sources that generate different traffic patterns, specific to applications such as file download, web browsing, video and voice.

# Appendix

**Theorem 1.** *The system is stable with the proposed $K_p$ and $K_i$ configuration.*

*Proof.* The closed-loop transfer function of our system is

$$\begin{aligned} S(z) &= \frac{-C(z)H(z)}{1-C(z)H(z)} = \\ &= \frac{-z(z-1)HK_p - zHK_i}{z^2 + (-HK_p - 1)z + H(K_p - K_i)} \end{aligned} \tag{5.1}$$

where

$$H = -\frac{\tau_{opt} p_{opt}(1 + p_{opt} \sum_{i=0}^{m-1} (2p_{opt})^i)}{2} \tag{5.2}$$

A sufficient condition for stability is that the poles of the above polynomial fall within the unit circle $|z| < 1$. This can be ensured by choosing coefficients $\{a1, a2\}$ of the characteristic polynomial that belong to the stability triangle [1]:

$$a_2 < 1 \tag{5.3}$$

$$a_1 < a_2 + 1 \tag{5.4}$$

$$a_1 > -1 - a_2 \tag{5.5}$$

In the transfer function of Eq. (5.1) the coefficients of the characteristic polynomial are

$$a_1 = -HK_p - 1 \tag{5.6}$$

$$a_2 = H(K_p - K_i) \tag{5.7}$$

From Eqs. (3.42) and (5.2) we have

$$HK_p = -0.4\frac{\tau_{opt}}{p_{opt}} \tag{5.8}$$

and from Eqs. (3.43) and (5.2) we have

$$HK_i = -\frac{0.4}{0.85 \cdot 2}\frac{\tau_{opt}}{p_{opt}} \tag{5.9}$$

from which

$$a_1 = 0.4\frac{\tau_{opt}}{p_{opt}} - 1 \tag{5.10}$$

$$a_2 = -0.16\frac{\tau_{opt}}{p_{opt}} \tag{5.11}$$

Given $\tau_{opt} \leq p_{opt}$, it can be easily seen that the above $\{a1, a2\}$ satisfy the conditions of Eqs. (5.3), (5.4) and (5.5). The proof follows. □

# References


[1] K. Aström and B. Wittenmark. *Computer-controlled systems, theory and design*. Prentice Hall International Editions, 2nd edition, 1990.

[2] A. Banchs, A. Azcorra, C. Garcia, and R. Cuevas. Applications and Challenges of the 802.11e EDCA mechanism: An Experimental Study. *IEEE Network*, 19(4):52–58, July 2005.

[3] A. Banchs and X. Perez. Distributed Fair Queuing in IEEE 802.11 Wireless LAN. In *Proceedings of IEEE ICC 2002*, New York, USA, April 2006.

[4] A. Banchs, X. Pérez-Costa, and D. Qiao. Providing Throughput Guarantees in IEEE 802.11e Wireless LANs. In *Proceedings of the 18th International Teletraffic Congress (ITC18)*, Berlin, Germany, September 2003.

[5] A. Banchs, P. Serrano, and H. Oliver. Proportional Fair Throughput Allocation in Multirate 802.11e EDCA Wireless LANs. *Wireless Networks*, 13(5), October 2007.

[6] A. Banchs and L. Vollero. Throughput Analysis and Optimal Configuration of 802.11e EDCA. *Computer Networks*, 50(11), August 2006.

[7] G. Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, March 2000.

[8] G. Bianchi, L. L. Fratta, and M. Oliveri. Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs. In *Proceedings of the Seventh IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC96)*, Taipei, Taiwan, October 1996.

[9] L. Bononi, M. Conti, and E. Gregori. Runtime optimization of ieee 802.11 wireless lans performance. *IEEE Trans. Parallel Distrib. Syst.*, 15(1):66–80, 2004.

[10] F. Calì, M. Conti, and E. Gregori. Dynamic tuning of the ieee 802.11 protocol to achieve a theoretical throughput limit. *IEEE/ACM Trans. Netw.*, 8(6):785–799, 2000.

[11] H. S. Chhaya and S. Gupta. Performance modeling of asynchronous data transfer methods of ieee 802.11 mac protocol. *Wirel. Netw.*, 3(3):217–234, 1997.

[12] G. F. Franklin, J. D. Powell, and M. L. Workman. *Digital Control of Dynamic Systems*. Addison-Wesley, 2nd edition, 1990.

[13] J. Freitag, N. L. S. da Fonseca, and J. F. de Rezende. Tuning of 802.11e Network Parameters. *IEEE Communications Letters*, 10(8):611–613, August 2006.

[14] M. Heusse, F. Rousseau, R. Guillier, and A. Duda. Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless lans. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 121–132, New York, NY, USA, 2005. ACM.

[15] M. Heusse, F. Rousseau, R. Guillier, and A. Duda. Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless lans. *SIGCOMM Comput. Commun. Rev.*, 35(4):121–132, 2005.

[16] C. Hollot, V. Misra, D. Towsley, and W.-B. Gong. A Control Theoretic Analysis of RED. In *Proceedings of IEEE INFOCOM 2001*, Anchorage, Alaska, April 2001.

[17] IEEE 802.11 WG. *Information Technology - Telecommun. and Information Exchange between Systems. Local and Metropolitan Area Networks. Specific Requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer Extension in the 2.4 GHz Band*. Supplement to IEEE 802.11 Standard, September 1999.

[18] IEEE 802.11 WG. *Information Technology - Telecommun. and Information Exchange between Systems. Local and Metropolitan Area Networks. Specific Requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*. Standard, IEEE, August 1999.

[19] IEEE 802.11 WG. *Amendment to Standard for Information Technology. LAN/MAN Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*. Supplement to IEEE 802.11 Standard, November 2005.

[20] IEEE 802.11 WG. *Information Technology - Telecommunications and Information Exchange between Systems. Local and Metropolitan Area Networks. Specific Requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*. IEEE 802.11-REVma/D9.0, Revision of Std. 802.11-1999, 2006.

[21] A. Nafaa, A. Ksentini, A. A. Mehaoua, B. Ishibashi, Y. Iraqi, and R. Boutaba. Sliding Contention Window (SCW): Towards Backoff Range-Based Service Differentiation over IEEE 802.11 Wireless LAN Networks. *IEEE Network*, 19(4):45–51, July 2005.

[22] Q. Ni, I. Aad, C. Barakat, and T. Turletti. Modeling and Analysis of Slow CW Decrease for IEEE 802.1 1 WLAN. In *Proceedings of the Seventh IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2003)*, Beijing, China, September 2003.

[23] Q. Pang, S. C. Liew, J. Y. B. Lee, and V. C. M. Leung. Performance evaluation of an adaptive backoff scheme for WLAN. *Wireless Communications & Mobile Computing*, 4(8):867–879, December 2004.

[24] N.-O. Song, B.-J. Kwak, J. Song, and M. E. Miller. Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease backoff algorithm . In *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference (VTC 2003-Spring)*, Jeju, Korea, April 2003.

[25] Q. Xia and M. Hamdi. Contention Window Adjustment for IEEE 802.11 WLANs: A Control-Theoretic Approach. In *Proceedings of IEEE ICC 2006*, Istanbul, Turkey, June 2006.

[26] Y. Yang, J. J. Wang, and R. Kravets. Distributed Optimal Contention Window Control for Elastic Traffic in Single-Cell Wireless LANs . *IEEE/ACM Transactions on Networking*, 15(6):1373–1386, December 2007.