

T.C.
GEBZE INSTITUTE OF TECHNOLOGY
GRADUATE SCHOOL OF ENGINEERING AND SCIENCES

**AN EFFICIENT WLAN CHANNEL UTILIZATION
WITH COLLISION-FREE ACCESS AND SMART
CHANNEL SCANNING PARADIGMS**

MEHMET FATİH TÜYSÜZ
PHD THESIS
DEPARTMENT OF COMPUTER ENGINEERING

GEBZE
2013

T.C.
GEBZE INSTITUTE OF TECHNOLOGY
GRADUATE SCHOOL OF ENGINEERING AND SCIENCES

**AN EFFICIENT WLAN CHANNEL UTILIZATION
WITH COLLISION-FREE ACCESS AND SMART
CHANNEL SCANNING PARADIGMS**

MEHMET FATİH TÜYSÜZ
PHD THESIS
DEPARTMENT OF COMPUTER ENGINEERING

SUPERVISOR
ASSOC. PROF. HACI ALİ MANTAR

GEBZE
2013



**GEBZE YÜKSEK TEKNOLOJİ
ENSTİTÜSÜ**

DOKTORA JÜRİ ONAY FORMU

GYTE Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 10/06/2013 tarih ve 2013/30 sayılı kararıyla oluşturulan jüri tarafından 26/06/2013 tarihinde tez savunma sınavı yapılan Mehmet Fatih TÜYSÜZ'ün tez çalışması Bilgisayar Mühendisliği Anabilim Dalında DOKTORA tezi olarak kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI)

: Doç. Dr. Hacı Ali MANTAR

ÜYE

: Prof. Dr. İbrahim SOĞUKPINAR

ÜYE

: Prof. Dr. Fatih ALAGÖZ

ÜYE

: Yrd. Doç. Dr. Hasari ÇELEBİ

ÜYE

: Yrd. Doç. Dr. Tansal GÜÇLÜOĞLU

ONAY

GYTE Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun tarih ve/..... sayılı kararı.

MZA/MÜHÜR

SUMMARY

IEEE 802.11 WLANs have reached an important stage and become a common technology for wireless access due to its low cost, ease of deployment and mobility support. In parallel with the extensive growth of WLANs, wireless stations nowadays require more efficient channel utilization than ever. Therefore, development of an efficient Medium Access Control (MAC) protocol that provides high throughput and Quality of Service (QoS) support has become a major focus in WLAN research. The IEEE 802.11 Distributed Coordination Functions (DCF/EDCA) are the fundamental MAC techniques of the IEEE 802.11 based WLAN standard, and they provide contention-based distributed channel access mechanisms for stations to share the wireless medium. However, performance of these mechanisms drops dramatically due to high collision probabilities as the number of active stations increases.

Towards closing the drawback of the abovementioned standards and increasing the wireless channel efficiency, this thesis proposes two novel adaptive collision-free MAC adaptations and a smart selective channel scanning scheme. Proposed MAC adaptation schemes prevent collisions and allow stations to enter the collision-free state regardless of the traffic load (saturated or unsaturated) and the number of stations on the medium. Additionally, in the proposed smart selective channel scanning scheme, periodic channel scanning is scheduled based on the predefined channel scanning intervals and activated after a successful frame transmission only if the mobile station is on move and the received signal strength indicator value (RSSI) drops below a pre-specified threshold. Analytical and simulation results show that the proposed schemes dramatically enhance the overall throughput, support QoS for real-time services and reduce the channel scanning overhead in IEEE 802.11-based WLANs.

Keywords: IEEE 802.11 WLANs; MAC; CSMA/CA; TDMA; QoS

ÖZET

IEEE 802.11 kablosuz yerel ağları (WLANs) düşük maliyeti, kolay kurulumu ve mobilizesinden dolayı günümüzde yaygın olarak kullanılmaktadır. Kablosuz ağ teknolojilerinin kısa surede hızlı yayılımı ile günümüzde kullanıcılar kanal verimi yüksek erişim teknolojilerine ihtiyaç duymaktadır. Bundan dolayı, performansı yüksek ve servis kalitesi (QoS) desteği sağlayabilen verimli bir Ağ Erişim Kontrol (MAC) protokolü tasarımı önemli bir araştırma konusu haline gelmiştir. IEEE 802.11 Dağıtık Koordinasyon Fonksiyonları (DCF/EDCA) istasyonların ağı paylaşımı için içerik temelli dağıtık kanal erişim mekanizmaları sağlar. Yalnız, bu mekanizmaların performansı ağdaki kullanıcı sayısının artması ve dolayısıyla paketlerin çarpışma ihtimalinin artmasına bağlı olarak önemli ölçüde düşer.

Yukarıda zikredilen kablosuz ağ erişim standartlarının dezavantajını gidermek ve kanal verimliliğini artırmak için bu tez kapsamında iki yeni uyarlamalı çarpışmasız ağ erişim adaptasyonları ve akıllı bir kanal tarama yöntemi sunulmaktadır. Önerilen ağa erişim adaptasyonları ağdaki çarpışmaları önler ve istasyonların trafik tipinden ve istasyon sayısından bağımsız olarak çarpışmasız bir faza geçmelerini sağlar. Tez kapsamında sunulan akıllı kanal tarama yönteminde ise, periyodik kanal tarama işlemi sadece istasyonun hareket halinde olması ve sinyal gücünün eşik bir değerin altına inmesi durumunda, istasyonun başarılı bir iletimi sonrasında belirlediği bir süre aralığında gerçekleştirilir. Analitik ve benzetim sonuçları tez kapsamında önerilen metotların IEEE 802.11 temelli kablosuz ağlardaki toplam veri hacmini ve servis kalitesini önemli ölçüde artırdığını, kanal tarama yükünü ise azalttığını göstermektedir.

Anahtar Kelimeler: IEEE 802.11 WLANs; MAC; CSMA/CA; TDMA; QoS

ACKNOWLEDGEMENTS

First of all I would like to thank to my supervisor Hacı Ali MANTAR, who has been a great mentor throughout my graduate studies. I will always be grateful to him for giving me this opportunity. Without his guidance and expertise, it would have been impossible to develop this thesis.

I am also thankful to M. Rasih Çelenliođlu and Gökhan Çelik, with whom I work together in the Computer Networks Laboratory, for their assistance in developing the simulation environment for the algorithms implemented through the thesis. All my colleagues in the department of Computer Engineering at Gebze Institute of Technology, especially Alper UĞUR, also deserve a thank you for their unending support and assistance.

Last but not least, I want to express my greatest gratitude to my family and my wife Ayşe Hümeýra Tüýsüz. Without their unconditional support, affection and encouragement, I am sure I would not be here right now.

TABLE OF CONTENTS

	<u>Sayfa</u>
SUMMARY	iv
ÖZET	v
ACKNOWLEDGEMENTS	vi
TABLE OF CONTENTS	vii
LIST OF ACRONYMS AND ABBREVIATIONS	ix
LIST OF FIGURES	xi
LIST OF TABLES	xiii
1. INTRODUCTION	1
2. BACKGROUND	6
2.1. IEEE 802.11 DCF and EDCA	6
2.1.1. Distributed Coordination Function (DCF)	6
2.1.2. Enhanced Distributed Coordination Function (EDCA)	7
2.2. IEEE 802.11 Handover Procedure	9
2.3. Beacon Frame	11
3. RELATED WORKS	12
4. UNINTERRUPTED COLLISION-FREE MAC ADAPTATION	17
4.1. Computing the Deterministic Value $V(d)$	21
4.2. Entrance of New Nodes and Maintaining the Collision-Free State	31
4.3. Adaptation to the EDCA: Traffic Prioritization	34
4.4. Fairness and Additional Observations	35
5. BEACON-BASED COLLISION-FREE CHANNEL ACCESS	37
5.1. Expected Convergence Time	40
5.2. Entrance of New Nodes and Maintaining the Collision-free State	45
5.3. Collision-free Adaptation of Unsaturated Stations	47
5.4. Adaptation to the EDCA: Traffic Prioritization	47
5.5. Coexistence of BCCA and DCF	48
6. SMART CHANNEL SCANNING WITH MINIMIZED SERVICE DISRUPTIONS	49
7. EVALUATION OF THE PROPOSED SCHEMES	56

7.1. Evaluation of Collision-free Channel Access - UCFA	56
7.2. Evaluation of Beacon-based Channel Access - BCCA	71
7.3. Evaluation of Smart Channel Scanning Scheme	80
7.4. Coexistence Evaluation of the Proposed Schemes	86
7.5. Energy-centric and Process-based Evaluation	86
8. CONCLUSION	91
REFERENCES	93
RESUMÉ	99
APPENDIX	100

LIST OF ACRONYMS AND ABBREVIATIONS

<u>Acronyms/ Abbreviations</u>	<u>Descriptions</u>
AC :	Access Category
ACK :	Acknowledgement
AIFS :	Arbitrary Interframe Space
AP :	Access Point
ARMA :	Auto Regressive Moving Average
BAT :	Beacon Arrival Time
BCCA :	Beacon-based Collision-free Channel Access
BE :	Best Effort
BEB :	Binary Exponential Backoff
BI :	Beacon Interval
BK :	Background
BS :	Base Station
CA :	Collision Avoidance
CSMA :	Carrier Sense Multiple Access
CSMA/CA:	Carrier Sense Multiple Access/Collision Avoidance
CTS :	Clear To Send
CW :	Contention Window
DCF :	Distributed Coordination Function
DCR :	Distributed Collision Resolution
DIFS :	Distributed Interframe Space
EDCA :	Enhanced Distributed Coordination Access
EKF :	Extended Kalman Filter
EU :	End User
GPS :	Global Positioning System
GSM :	Global System for Mobile
HCL :	Hit Channel List
HHO :	Horizontal Handover
HO :	Handover
IEEE :	Institute of Electrical and Electronics Engineers

IP	:	Internet Protocol
IS	:	Information Server
L-BEB	:	Learning Binary Exponential Backoff
MAC	:	Media Access Control
MIH	:	Media Independent Handover
MT	:	Mobile Terminal
NAV	:	Network Address Vector
NIC	:	Network Interface Card
PoA	:	Point of Attachment
PREQ	:	Probe Request
PRES	:	Probe Response
QoS	:	Quality of Service
RC	:	Retransmission Count
RSS	:	Received Signal Strength
RSSI	:	Received Signal Strength Indicator
RTS	:	Request To Send
SIFS	:	Short Interframe Space
SNR	:	Signal to Noise Ratio
SRB	:	Semi Random Backoff
SSID	:	Service Set Identifier
TCP	:	Transmission Control Protocol
TDMA	:	Time Division Multiple Access
TXOP	:	Transmission Opportunity
UCFA	:	Uninterrupted Collision-free Channel Access
UDP	:	User Datagram Protocol
VHO	:	Vertical Handover
VI	:	Video
VO	:	Voice
VoIP	:	Voice over Internet Protocol
WLAN	:	Wireless Local Area Network
ZC	:	Zero Collision

LIST OF FIGURES

<u>Figure No:</u>	<u>Page</u>
2.1: IEEE 802.11 MAC access mechanisms.	7
2.2: The timing relationship for EDCA.	9
2.3: IEEE 802.11 active scanning handover procedure.	10
4.1: UCFA deterministic backoff selection procedure.	18
4.2: Binary Exponential Backoff procedure in CSMA/CA CSMA/CA.	19
4.3: Backoff countdown procedure of an unsaturated station.	20
4.4: Collision probability vs. Number of stations.	24
4.5: Accuracy of the ARMA and Kalman filters for the Standard DCF and the DCF-UCFA mixed scenario.	29
4.6: Procedure of a new entrant in UCFA.	33
5.1: BCCA deterministic backoff selection procedure.	38
5.2: Binary Exponential Backoff procedure in CSMA/CA.	39
5.3: Maximum number of gaps for various numbers of stations.	44
6.1: Flowchart of the proposed smart channel scanning scheme.	52
6.2: Smart channel scanning procedure of a VoIP station.	55
7.1: IEEE 802.11 network communication scenario.	57
7.2: Throughput values of related schemes when the channel is ideal and all stations are saturated.	58
7.3: Throughput values of related schemes when the traffic is a mix of voice, video and data stations.	60
7.4: Throughput values of related schemes. 3 new stations enter the medium after the 50 th second.	62
7.5: Average throughput vs. Number of Hidden nodes for related schemes.	63
7.6: Coexistence throughputs of UCFA-DCF, ZC-DCF, L-BEB-DCF and DCF itself when channel error-rate is 1%.	64
7.7: Fairness & average throughputs for DCF, UCFA and Coexistent stations.	66
7.8: Channel efficiency vs. Number of stations when $V(d) = 16$.	67
7.9: Coexistence throughputs of related schemes when $V(d) = 16$ and Nr. of voice, data, video stations are 6, 4, 6.	69
7.10: Collision and Delay variation values for the Coexistent scenario.	70

7.11: Channel efficiency vs. Number of stations. BCCA has an adaptive cycle length L-BEB and ZC have a fixed cycle length; $V(d) = 16$.	72
7.12: Number of expected cycles for Collision-free state and Stable-state.	74
7.13: Throughput values of related schemes.	75
7.14: Throughput values of related schemes.	77
7.15: Coexistence throughput values of BCCA-DCF, ZC-DCF, L-BEB-DCF and DCF itself.	78
7.16: Number of collisions vs. Time for the coexistence situation.	80
7.17: IEEE 802.11 network communication scenario.	81
7.18: Channel scanning comparison of related schemes.	82
7.19: Channel scanning comparison of related schemes.	84
7.20: Throughput values of related schemes.	85
7.21: Channel efficiency vs. Number of stations for UCFA and BCCA.	87
7.22: Coexistence situation throughput performance of UCFA and BCCA when the smart channel scanning scheme is on and off.	88

LIST OF TABLES

<u>Table No:</u>	<u>Page</u>
2.1: Default EDCA parameters	8
7.1: Timing units and values for IEEE 802.11b	56
7.2: IEEE 802.11b parameter values used at simulations	56
7.3: Timing units and values for IEEE 802.11b	81
7.4: IEEE 802.11b parameter values used at simulations	81

1. INTRODUCTION

A Wireless Local Area Network (WLAN) is a computer network that uses the IEEE 802.11 wireless communication standard to connect computers and devices in a limited geographical area, such as house, school, work place and etc. It typically extends the coverage and mobility of an existing wired Local Area Network (LAN) by attaching a device, called Access Point (AP), to the edge of the wired network. Users communicate with the AP using a wireless network adapter in a similar way to the traditional Ethernet adapter. This enables users to move around within vicinity and still be connected to the network. Therefore, WLAN often provides the last mile wireless access to the wired network.

In IEEE 802.11 WLANs, access to the communication channel is regulated by the Medium Access Control (MAC) protocol. MAC plays an important role in determining the channel utilization since it is designed for sharing a single and limited channel among stations. Efficient MAC algorithms are crucial in minimizing collisions between transmissions and thus achieving high utilization of the channel. The Distributed Coordination Function (DCF) is the basic channel access method in the IEEE 802.11 standard. It employs a carrier sense multiple access with collision avoidance (CSMA/CA) algorithm and a Binary Exponential Backoff (BEB) procedure.

The Channel Access procedure in WLANs can be classified into two broad categories, namely, Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and Time Division Multiple Access (TDMA). They are two well-known medium access mechanisms for shared medium communication systems. In CSMA/CA, a carrier sensing scheme is used. A station ready to transmit its data has to listen to the channel first for a certain amount of time to determine whether another station is transmitting on the channel within the wireless range. There is no tight time synchronization in CSMA/CA and stations simply transmit when they have data to transmit. Collision avoidance is used to improve the CSMA performance by assigning random backoff values to stations. Thus, CSMA/CA reduces the probability of collision due to the use of a random binary exponential backoff (BEB) time. However, CSMA/CA still cannot recover from collisions entirely and hence; it may cause inefficient use of the medium. TDMA, on the other

hand, allows several users to share the same frequency channel by dividing the signal into different time slots. Stations transmit in rapid succession, one after the other, each using its own time slot. This procedure allows multiple stations to share the same transmission medium while using only a part of its channel capacity. However, TDMA requires tight time synchronization among participating stations. Additionally, a prior setup is required to assign slots to the active stations. This set-up causes extra signaling overhead and often requires a central decision point [Alonso *et al.*, 2006].

Both CSMA and TDMA have advantages and disadvantages. CSMA/CA is fairly simple to implement but collisions may cause inefficient use of the medium. TDMA eliminates the possibility of collisions. For QoS-aware applications, TDMA is usually more appropriate than contention-based CSMA protocols since it may achieve higher channel efficiency avoiding collisions. However, set-up is required and a central unit has to maintain detailed knowledge of each station in TDMA, which requires additional data exchange.

Recently, the implementation of a new distributed collision-free WLAN MAC adaptation that combines the advantages of both CSMA/CA and TDMA has taken an important place in the literature. Borrowing the idea of reservation from TDMA and applying it in wireless CSMA/CA networks can enhance the performance of DCF and EDCA. However, achieving the reservation in the context of CSMA networks is challenging due to its different nature from TDMA networks and the random structure of BEB procedure.

Additionally, due to the design choices and requirements, the communication range of the IEEE 802.11 standard is limited. Therefore, WLAN coverage over a large area is provided using multiple APs. Due to the mobility of mobile stations, a station might move and lose the signal of its associated AP. In that case, the station would try to maintain its wireless connectivity by associating with a new AP in a process known as handover.

When a mobile station decides to handover, it first discovers the neighboring APs. Cali *et al.* show that discovery phase accounts for more than 90% of the total handover latency. Besides, empirical analysis shows that potential handover latency is around 300 ~ 500 milliseconds [Sakurai *et al.*, 2007]. This high latency results in play-out gaps and disrupts the ongoing communications of delay-sensitive multimedia applications. Therefore, the elapsed time in the scanning phase is the

crucial problem for the fast seamless handover procedure.

In a handover process, it is not desirable for a station to wait for several seconds until the device has scanned all possible channels for available networks. Hence, most devices perform a periodic background channel scanning to check the current link quality and to obtain the information of the neighboring APs at regular intervals, typically in the range of 1, 5 and 10 minutes [IEEE Std. 802.21]. Periodic channel scanning would be very useful if the station is highly mobile. However, it causes significant overhead in a static environment. Therefore, unnecessary channel scanning of mobile stations degrades both the energy efficiency and the efficiency of the channel utilization.

Another drawback of the channel scanning is the interruption of ongoing communications of stations. It is because stations that have single radio interface cannot provide frame delivery and channel scanning simultaneously. Therefore, channel scanning procedure must be carefully scheduled together with the frame delivery process to minimize the interruption of ongoing communications. However, existing channel scanning schemes, such as *passive*, *active*, *selective*, and *unicast* scanning do not provide limited service interruption for stations.

Main contributions of this thesis can be grouped into two as; contributions related to increasing the wireless channel efficiency at frame delivery phases and contributions related to reducing the channel scanning interval of wireless stations.

Throughout the thesis, we first propose two novel adaptive and uninterrupted collision-free MAC adaptation schemes, in Sections 4 and 5, to improve the channel efficiency of the standard DCF/EDCA mechanisms. The proposed schemes are fundamentally different from the standard random binary exponential backoff procedure as they provide a form of resource reservation at no extra cost. The key ideas of the proposed schemes are to carefully control the backoff process and to make the best use of resources on medium. In the proposed schemes, certain number of time slots constitutes a backoff period. If a station successfully transmits its frame in a time slot, then the station deterministically uses the same time slot in consecutive backoff periods as in the TDMA.

In this context, we show how the proposed schemes can achieve and maintain its collision-free state for both saturated and unsaturated networks even if there are new stations that enter the medium. The proposed schemes can fairly coexists with the legacy CSMA/CA stations. It can be implemented without any changes on the

IEEE 802.11 standard. Each station makes use of its local information. Performances of the proposed schemes are investigated by numerical analyses and extensive simulations. Simulation results show that the proposed schemes are very effective and improve the channel performance under a wide range of contention levels.

This thesis also addresses four issues in Section 6 to reduce the channel scanning overhead and to provide fast handover. First, the proposed smart channel scanning scheme separates the background channel scanning from the handover procedure. Second, it reduces the size of the set of channels to be scanned. Third, it reduces the scanning time on each channel and finally, it minimizes ongoing communication interruptions.

In order to perform background channel scanning with minimized communication interruptions and to improve the throughput performance of stations, our proposed algorithm waits for the station to transmit its frame successfully before switching to another channel for the background scanning. After the successful frame delivery, the station switches to other channels and performs background scanning during a pre-computed interval, T_{scan} . With the end of this interval, the station switches back to its channel and continues to frame transmission and reception. Therefore, this procedure will minimize the communication interruptions of stations and improve the overall throughput with smart distribution of *channel-scanning* and *frame-delivery* phases.

Consequently, the main feature of the proposed smart channel scanning scheme is that the fast handover is accomplished without degrading the multimedia and data services during the handover procedure. The proposed model has been designed to provide higher user quality experience in accordance with the QoS requirements of multimedia applications, and to increase the overall throughput of saturated TCP stations. In this context, extensive simulations have been performed to demonstrate the efficiency of the proposed approach.

The rest of the thesis is organized as follows. Section 2 describes the operations of the IEEE 802.11 DCF and EDCA standards, IEEE 802.11 handover procedure and the beacon frame, respectively. Section 3 presents the related works and the motivation. Section 4 presents the concept of Uninterrupted Collision-Free MAC Adaptation, its application in the IEEE802.11 DCF/EDCA, computation of the deterministic backoff value, the coexistence situation of the proposed uninterrupted collision-free MAC adaptation with the standard DCF and maintaining the collision-

free state even if there are unsaturated stations on the medium or in case of new entrances, respectively. Section 5 presents the concept of Beacon-based Collision-Free MAC Adaptation, its application in the IEEE802.11 DCF/EDCA, expected convergence time to the collision-free state, fairness and the hidden node problem and maintaining the collision-free state even if there are unsaturated stations on the medium or in case of new entrances, respectively. Section 6 presents the concept of the proposed smart channel scanning scheme. Section 7 presents the evaluation process and the results of all the proposed schemes in this thesis. Finally, Section 8 reports the final evaluations.

2. BACKGROUND

Along the following subsections, a deep literature study is presented. In this chapter, all the concepts covered in this thesis are described in detail to provide the reader with a full understanding of the topic.

2.1. IEEE 802.11 DCF and EDCA

This subsection presents a summary of the standard DCF and EDCA since it is important to know these structures before paying attention to our contributions.

2.1.1. Distributed Coordination Function (DCF)

DCF is the fundamental MAC technique of the IEEE 802.11 based WLAN standard. The DCF process of a station is described as follows. When a station has a new frame to transmit, it monitors the channel. If the channel is idle for a period of time equal to Distributed Inter-Frame Space, DIFS, it starts transmitting. If the channel is sensed busy, it keeps monitoring the channel until it is sensed idle for a DIFS interval. After the DIFS interval, the backoff process starts. The initial backoff counter is a uniformly distributed random integer and has a predefined range $[0, CW]$. Stations initialize their backoff counters within this range. Contention Window (CW) value depends on the number of failed transmissions of a frame. CW is set equal to CW_{\min} at the first transmission attempt. As long as the channel is sensed idle, backoff counter is decremented once per slot time. Backoff counter is frozen whenever a transmission is sensed on the channel, and it is reactivated after the channel is sensed idle for a DIFS interval.

Whenever the backoff counter reaches zero, the station transmits its frame in the next slot time. When the frame is received successfully, an acknowledgement (ACK) is sent to notify transmitting station that the frame has been successfully received. If the ACK is not received within a given timeout, the station assumes that the frame was not received. Then, the station reschedules its transmission by re-entering backoff process. CW is roughly doubled ($CW = 2(CW + 1) - 1$) with each unsuccessful retransmission until it reaches CW_{\max} . If the number of retransmissions

reaches the maximum retransmission limit, then the frame is dropped. Either the frame is successfully received or finally dropped, the station resets its CW value to CW_{min} . Algorithm 1 summarizes the Binary Exponential Backoff procedure (BEB).

Algorithm 1: BEB (Binary Exponential Backoff)

```

if (  $R_c == 0$  ) then           // First transmission
     $CW = CW_{min}$ 
else                             // Retransmission
     $CW = 2(CW + 1) - 1$ 
     $CW = \min(CW, CW_{max})$ 
    Backoff =  $\text{rand}(0, CW) \times (\text{slot})$ 

```

where R_c is the retransmission count, which is set to 0 at the first transmission and incremented by one per retransmission. $\min(CW, CW_{max})$ returns the smaller number. $\text{rand}(0, CW)$ returns an integer value chosen randomly between 0 and CW.

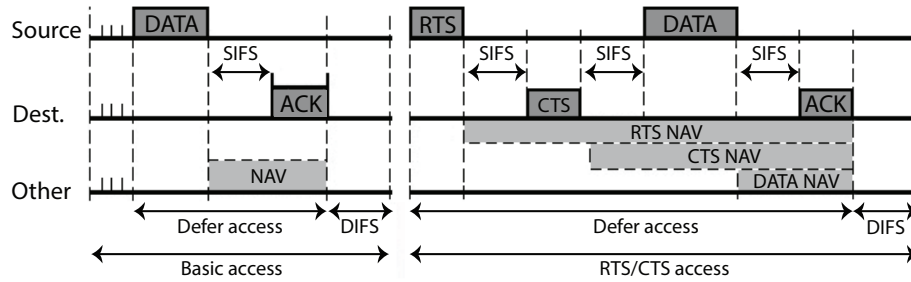


Figure 2.1: IEEE 802.11 MAC mechanisms.

As is shown in Figure 2.1, the DCF also has an optional virtual carrier sense mechanism that exchanges short Request-to-send (RTS) and Clear-to-send (CTS) frames between source and destination stations during the intervals between the data frame transmissions. RTS/CTS scheme can be used in WLANs to prevent collisions. However, the RTS/CTS procedure has an additional overhead on the channel that may penalize the performance. Therefore, the rest of the thesis focuses on the basic two-way handshaking mechanism.

2.1.2. Enhanced Distributed Coordination Function (EDCA)

The IEEE committee has also developed another access method called IEEE 802.11e EDCA [IEEE Std 802.11e-2005] to enhance the original IEEE 802.11 and to

support the required QoS for multimedia applications. It offers differentiated service by providing different waiting times for different traffic priority levels. EDCA classifies traffic into four Access Categories (AC): Voice (VO), Video (VI), Best-Effort (BE) and Background (BK). Voice is assigned as the highest priority while background traffic is the lowest. Table 2.1 outlines the summary of EDCA parameters.

Table 2.1: Default EDCA parameters.

Priority Level	AC	CW_{\min}	CW_{\max}	AIFSN
0	VO	7	15	2
1	VI	15	31	2
2	BE	31	1023	3
3	BK	31	1023	7

CW_{\min} and CW_{\max} determine the upper and lower bounds of contention window for each access category (AC). These two variables adjust the range of the random backoff values. In other words, they adjust the random waiting times before accessing the medium. Different CW ranges are assigned for different ACs. The idea behind CW range is to give small CW values to the high priority traffic and hence, to have a short waiting time before accessing the medium. Figure 2.2 shows the timing relationship of the EDCA. In EDCA, if the channel is idle for a period of time equal to Arbitrary Inter-Frame Space, *AIFS*, it starts transmitting. *AIFS* is a method of prioritizing one access class over the other by modifying the time of stations that have to listen to an idle channel before decrementing the backoff counter. Additionally, EDCA provides contention-free access to the channel for a period called a Transmit Opportunity (TXOP). When *TXOP* is used, stations are allowed to transmit multiple frames.

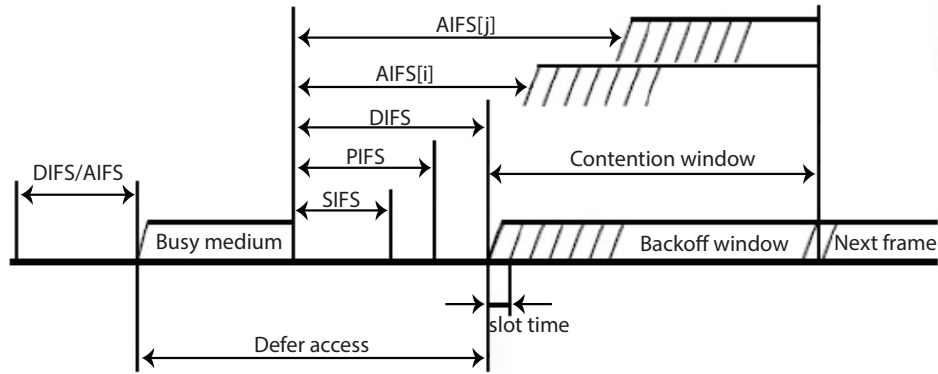


Figure 2.2: The timing relationship for EDCA.

Consequently, stations choose small CW values after successful transmissions and large CW values after unsuccessful transmissions. Real-time applications such as VoIP, video-conferencing or online games are delay-sensitive multimedia applications and require relatively small delay and small delay variation. The IEEE 802.11e standard (EDCA) has the ability to accommodate real-time traffic and to provide relatively small delay. However, it cannot achieve small delay variation due to the burst feature of its backoff mechanism after a collision or a frame error [Tuysuz *et al.*, 2011].

2.2. IEEE 802.11 Handover Procedure

The process of transferring an ongoing call or data session from one AP to another is known as handover. Handover is initiated whenever the received signal strength indicator (RSSI) of the current AP drops below a pre-specified threshold value. Handover procedure can be divided into three phases: *discovery*, *re-authentication* and *re-association*. The methodology used for a handover is as follows. Whenever a station wants to access a channel, the station needs to get synchronization information from the AP. There are two ways (*passive* and *active scanning*) to get this information and it is called *discovery phase*. In the *passive scanning*, stations wait for beacon frames broadcasted periodically by APs at each channel. Beacon interval is typically configured as 100ms. Therefore, it takes a considerable time to scan all the channels with *passive scanning*. In the *active scanning*, stations first transmit *Probe Request Frames* and wait for the *Probe Response Frames* from the APs. This frame exchange allows stations to consider basic information about the channel, such as the signal strength, available

transmission modes, etc. Frame exchange is repeated for all APs in the vicinity and hence, stations would have a list of APs at the end of the scanning process. Figure 2.3 summarizes the handover procedure.

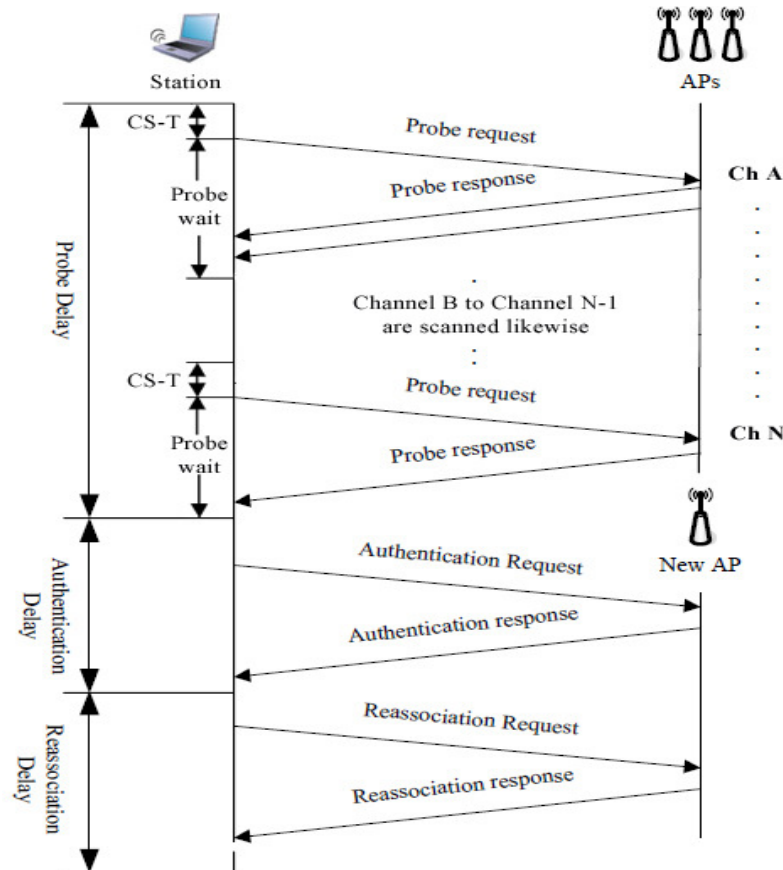


Figure 2.3: IEEE 802.11 active scanning handover procedure.

where *CS-T* (Channel Switching and Transmission) is the time to switch and transmit the *probe request frame* on a channel. *Probe wait* is the time to spend on the channel to gather all the *probe responses* from AP(s) operating on that channel. When the scanning procedure is over, *re-authentication* phase starts. Authentication is a necessary prerequisite to associate to an AP. In this phase, station authenticates with the most proper AP in the *discovery phase*. *Re-association phase* is the process for transferring associations from one AP to another. Once the station has been authenticated with the new AP, the re-association can be started. At the end of this procedure, the station associates to the AP that has the maximum received signal strength indicator (RSSI).

Consequently, channel scanning is the dominant factor that mostly affects the handover delay and hence, it has to be reduced to provide seamless handover.

2.3. Beacon Frame

Beacon frame is one of the management frames in IEEE 802.11 based WLAN standard. It contains all the information about the network and enables stations to establish and maintain communications in an orderly fashion. In an infrastructure BSS, beacon frames are transmitted by the Access Point (AP) periodically to announce the presence of a wireless LAN. A typical beacon frame carries the following information in the frame body,

- Beacon interval: This is the time interval between beacon transmissions. Typically configured as 100ms in the AP.
- Timestamp: After receiving a beacon frame, a station uses the timestamp value to update its local clock. This process enables synchronization among all stations in the AP.
- Service Set Identifier (SSID): The SSID identifies a specific wireless LAN. Before associating with a particular WLAN, a station must have the same SSID as the AP.
- Supported rates: Each beacon carries information that describes the rates that the particular WLAN supports. With this information, stations can use the performance metrics to decide which AP to associate with.

There are no reservations for sending beacons, and they must be sent using the 802.11 CSMA/CA protocol. If another station is sending a frame when the beacon is to be sent, then the AP must wait. As a result, the actual time between beacons may be longer than the beacon interval. Stations, however, compensate for this inaccuracy by utilizing the timestamp found within beacon. As you can see, beacons are very important. Without them, a wireless LAN simply will not work.

3. RELATED WORKS

A great variety of medium access control (MAC) schemes have been developed and studied for wireless communication systems in the literature. Some of them are related to the implementation of an efficient solution that handles the collisions. To this end, many collision resolution algorithms have been proposed, such as the work in [Kwon *et al.*, 2004]. This work is based on increasing the contention window exponentially and generating a new backoff counter when a station detects a busy period. In case a station detects a number of consecutive idle slots, then it reduces the backoff counter exponentially. In [Haas *et al.*, 2003], an improved exponential backoff algorithm, which adjusts the increasing and decreasing factors of the contention window and backoff value is proposed in a similar fashion. Sun *et al.* propose a distributed contention-based MAC algorithm, called distributed collision resolution (DCR) scheme. DCR can provide high throughput, low delay and delay variation while maintaining the basic contention window and binary exponential backoff procedure of the IEEE 802.11 DCF. However, all the proposed works above always make use of randomly chosen backoff values without any knowledge of the medium. Therefore, they still have collision problems.

Heusse *et al.* propose a novel access method derived from 802.11 DCF in which all hosts use similar values of the contention window (CW) to benefit from good short-term access fairness. The method is called *Idle Sense*, since each host observes the mean number of idle slots between transmission attempts to dynamically control its contention window. *Idle Sense* maintains good aspects of the DCF, such as “*no explicit information exchange*” and modifies the weak aspects, such as “*exponential backoff procedure*”. Therefore, it improves the performance of the standard DCF in terms of throughput and fairness. However, its random slot allocation mechanism does not prevent collisions. If a collision or a frame error occurs during transmission, then there will be station(s) selecting a random backoff value before re-attempting to transmit. These stations may cause new collisions and may initiate a chain reaction, which brings the system to its transitory CSMA-like operation.

New hybrid MAC protocols that are composed of TDMA and CSMA/CA have been proposed recently. Alonso *et al.* propose a hybrid protocol that is called

Distributed Queuing Random Access Protocol (DQRAP/CDMA) when used in a CDMA environment, and also called Distributed Queuing Collision Avoidance (DQCA) when used in a WLAN TDMA environment. DQRAP/DQCA behaves as a random access mechanism for low traffic load and switches smoothly and automatically to a reservation scheme when traffic load grows. Thus, the best of each mechanism is retained. Barcelo *et al.* propose CSMA with Enhanced Collision Avoidance (CSMA/ECA), primarily called L-BEB that behaves as the standard CSMA/CA protocol with one exception, which is the selection of a deterministic backoff value after successful transmissions. Learning Binary Exponential Backoff (L-BEB) uses a fixed or reselected random backoff values to achieve collision-avoidance. As with IEEE 802.11 DCF, it chooses the backoff values based on the success or failure of the last transmission. Therefore, it is easy to implement on existing platforms. Yong *et al.* present the Semi Random Backoff procedure (SRB). This is similar to the L-BEB as it also uses a fixed backoff value to achieve collision-avoidance. Although both L-BEB and SRB employ deterministic backoff selection in a similar way to resolve network collisions, their analytic models, verification approaches and evaluation procedures are different. In case of failed transmissions, SRB automatically moves back to the standard random backoff procedure as the L-BEB does.

CSMA/TDMA hybrid approaches use a deterministic backoff value after each successful transmission as long as the medium does not suffer from a collision. Whenever a collision occurs, collided stations move back to their random behavior as the IEEE 802.11 DCF. Therefore, in case the number of new entrants is high, performance of the hybrid schemes will be closer to the CSMA performance and the advantages of this kind of algorithms such as L-BEB or SRB will not be so obvious. As a result, instead of selecting a random backoff value after a collision or a frame error, an efficient and adaptive approach, which logically selects one of the possibly empty slots in a backoff period, is required.

An approach that works based on the empty and busy slots in the network, namely Zero-collision (ZC) is proposed in [Lee *et al.*, 2008]. ZC is a decentralized scheme that achieves fast convergence to the collision-free state. Instead of observing only the slot it transmits in, ZC observes each of MAC slots separately. ZC is similar to L-BEB since it effectively chooses a deterministic backoff after a successful transmission. However, ZC converges to the collision-free state faster

than L-BEB or SRB since the procedure after a collision is different. In ZC, a station observes the occupancy of each slot in the previous period. The station chooses its new backoff value uniformly between the slot it failed and the slots that were idle in the last period. Therefore, converging to the collision-free state becomes faster than other schemes since ZC avoids other busy slots, which other stations may have ‘reserved’. Fang *et al.* propose two modifications to L-BEB and ZC in order to provide a good approximation to the ideal hybrid MAC. In this context, authors propose a fully decentralized Learning MAC (L-MAC) that uses the same information as L-BEB, but achieves faster convergence. In a similar way, they demonstrate a learning version of ZC (L-ZC) that provides convergence faster than ZC.

Fang *et al.* also propose a new method for unsaturated stations. In this method, stations update their schedules not to forget the exact slots they transmit. Whenever a station has a new frame to transmit, if the exact slot it transmitted earlier is still idle, the station selects the same time slot and transmits in that slot. However, if the slot is already captured by another station, then the proposed method may allow a station to select a slot that is actually reserved by another station but not used at that time due to the nature of unsaturated transmission. Therefore, even in the presence of the method proposed in [Fang *et al.*, 2013], unsaturated stations may collide if they have a new frame in their queue to send and select the same slot, which seems empty at the same time. As a result of a new collision, collision-free state is broken. In this thesis, in order to prevent collisions of unsaturated stations and to maintain the collision-free state, we propose a new method called *Virtual frame initialization*. In this method, in order to increase the efficiency of slot selection, stations implemented with the proposed scheme maintain the status (busy or empty) of each slot in previous periods in their memories, classify them and finally, select one of the high priority slots for their next transmission attempts.

In all these hybrid works, the traffic is mainly assumed as saturated and channel error is mainly out of the scope. It is also assumed that the number of stations in the network is known or less than a deterministic backoff value (a mechanism that automatically adapts the schedule length is used in [Fang *et al.*, 2013]). In reality, all the factors explained above are common metrics that can significantly affect the performance of wireless networks. A solution that does not cover all these factors may result well for few cases in theory but not in practice.

Therefore, a revision for the TDMA/CSMA hybrid approaches, which takes all these factors explained above into account, is required.

In addition, there have been many works focusing on the handover process over IEEE 802.11 WLANs in the literature as well. Brief descriptions of well-known algorithms are as follows.

Received signal strength indicator (RSSI)-based handover is the most widely used handover method. The RSSI is directly related to the distance of the station to its AP and it is the easiest way to measure the service quality. Therefore, most of the existing handover algorithms use the RSSI as the main decision criterion. *Available bandwidth-based handover* is related to the traffic condition in the network. Available bandwidth is a bit/sec expression and it indicates the available data resources.

Power consumption-based handover is related to the battery level of mobile stations and aims to initiate a handover whenever it discovers a network that consumes less power [Kassar *et al.*, 2008]. Monetary cost, security or user preferences are also important parameters for the handover process. Some algorithms take the charging, security or user policies into consideration for different networks in making their handover decision [Wang *et al.*, 2008].

Cheng *et al.* propose an AP selection algorithm that estimates the traffic loads of APs by observing the IEEE 802.11 frame delays and use the results to determine which AP to handover. However, it performs too slow as its primary aim is the efficient distribution of the channel utilization. There have been also some works [Dong *et al.*, 2009], [Yu *et al.*, 2009] that make use of a central manager to reduce the scanning overhead of stations. These works build network graphs using the information, such as the neighboring AP list and their coordinates received from a server. These works may provide fast handover up to a point since they rely on the information received from a server but there is no effort made to enhance this information in mobile stations. In order to provide fast and seamless handover, Ramani *et al.*, propose a method called “Sync Scan”, which achieves fast handover by synchronizing scanning times of all APs. However, the AP synchronization is not an easy task and it also requires additional changes in the standard.

There are also many works that focus on reducing the scanning delay for a fast handover in the literature. Sooyong *et al.* address the scanning overhead and propose a smart scanning method that uses a low-power motion-sensor equipped in mobile

devices. However, this scheme interrupts ongoing communications of stations. It is because the scheme does not schedule the scanning intervals and scanning can be activated anytime the station is on move or the RSSI value of the associated AP drops below a pre-specified threshold.

The frequency of the periodic channel scanning also needs to be adjusted since it may cause significant scanning overhead in a static environment (as much as 13% of the entire traffic as mentioned in [Raghavendra, *et al.*, 2010]). In order to reduce the frequency of channel scanning, a few approaches that adaptively adjust the scanning interval were proposed in the literature.

Kim *et al.* propose a method that utilizes the Network Address Vector (NAV) for stations that are not involved in the transmission to schedule the pre-scan process. This method aims at minimizing communication interruptions of ongoing sessions by scanning the other channels only when the station is in the NAV interval and not involved with a frame delivery. However, NAV interval might be a very short duration, especially for delay-sensitive frame transmissions. Hence, this duration may not be enough even for scanning one channel.

Igor *et al.* propose a smart channel scanning scheme where the channel scanning is scheduled based on predicting the channel scanning time analytically. In this scheme, maximum service interruption time is a preconfigured value with the maximum value of 50ms, which is the maximum allowed delay to prevent excessive jitter. However, turning the channel off for 50ms does not mean that the frames will be sent to their destinations 50ms late. Frames that enter the contention 50ms late would be late more than 50ms since other stations also contend to access the channel. Therefore, this procedure will degrade the performance of delay-sensitive multimedia applications. Besides, waiting for 50ms also interrupts the ongoing communications of stations. Additionally, the authors also assume that APs can receive the information of neighboring networks from the information server (IS) defined by the IEEE 802.21 standard and broadcast this information to mobile stations via beacon frames.

There have been also some works [Shafer *et al.*, 2010] that make use of a central manager to reduce the scanning overhead of stations. These works build network graphs using the information, such as the neighboring AP list and their coordinates received from a server. As mentioned above, these works need a central manager and may require small changes in the IEEE 802.11 WLAN standard.

4. UNINTERRUPTED COLLISION-FREE MAC ADAPTATION

In CSMA/CA, an empty slot is much shorter than a successful (busy) slot¹. A slot containing a collision or an error is even longer than a successful slot. In short, an empty slot is much more desirable than a collision since the duration of an empty slot is much shorter than the duration of a collision slot [Calì *et al.*, 2000]. Therefore, in order to increase the MAC layer channel efficiency, it is critical to reduce the number of collisions first, and then to keep the number of empty slots relatively low.

The proposed scheme that we call *Uninterrupted Collision-Free MAC Adaptation* (UCFA) assumes that all stations on the medium are bounded by a deterministic backoff value², $V(d)$. In UCFA, if a station successfully transmits its frame in a time slot, then the backoff value of the station is deterministically assigned as $V(d)$. This procedure allows the station to use the same time slot in consecutive backoff periods as in TDMA. Since the selection of transmission slot is deterministic over a period, as soon as all stations successfully transmit and capture their slots, each of the stations periodically transmits in every $V(d)$ slot. Hence, collisions disappear, and stations orderly transmit in a TDMA fashion.

In UCFA, each station maintains and updates the set of empty slots in a distributed manner by observing the last $V(d)$ period. As it is known, a slot is empty if no transmission occurs in that slot and all the contending stations can detect that slot as an empty slot. In UCFA, stations note all the empty slots within the $V(d)$ period and make use of this information for the next $V(d)$ period. Since the schedules are periodic and have a length corresponding to a deterministic number of slots $V(d)$, no agreement is required on labeling of the slots. Therefore, reservation is a virtual process at no extra cost. In UCFA, each station randomly and uniformly selects one of the empty (non-reserved) slots within the previous $V(d)$ period for its next transmission attempt. As long as the transmission is successful, the station remembers that slot as being reserved by itself and uses the same deterministic backoff value $V(d)$ for its next transmission attempts. The procedure when the

¹e.g., the duration of an empty slot is 20 μ s in IEEE 802.11b while the duration of a successful slot is in the order of ms.

²Section 4–b presents the $V(d)$ computation process.

transmission is not successful due to a frame error or a simultaneous transmission of an entrant is explained in detail in the next sections.

Deterministic backoff selection process³ in UCFA and CSMA/CA are demonstrated in Figure 4.1 and Figure 4.2, respectively. In the figures, black boxes represent successful transmissions and striped boxes represent collisions. Three saturated stations are contending to access the channel. UCFA selects a deterministic backoff value after a successful transmission and a random value between empty slots in case of a collision. However, the CSMA/CA always chooses a random backoff value. In UCFA, after all participating stations successfully transmit, they acquire a particular slot and keep transmitting in that order. Therefore, the system enters its collision-free state. However, collisions may occur anytime in CSMA/CA due to the standard random binary exponential backoff procedure.

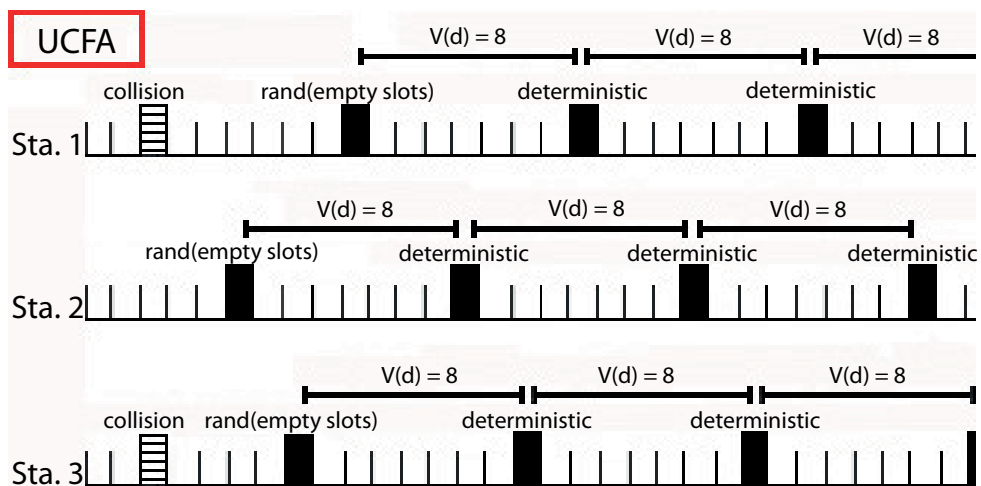


Figure 4.1: UCFA deterministic backoff selection

³Although the durations of empty, successful and collision slots are different, they are represented equally in the figures for simplicity.

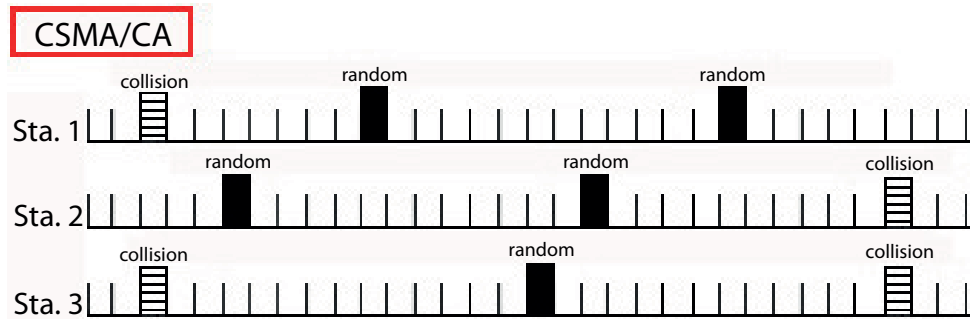


Figure 4.2: Binary Exponential Backoff procedure in CSMA/CA.

Existing hybrid approaches such as L-BEB and ZC do not provide an appropriate solution for unsaturated networks. In L-BEB, a deterministic backoff procedure cannot be used for unsaturated traffic since stations do not always have a frame to transmit. L-BEB uses the standard binary exponential backoff procedure for unsaturated networks. This leads to further collisions and may cause a similar performance to the standard CSMA/CA. The reaction of ZC adaptation is smarter than L-BEB. ZC randomly selects one of the empty slots in the previous period for unsaturated stations as well. However, in reality, there might be many stations using unsaturated traffic on the medium. Just because a station did not transmit in its previous period, it does not mean that it will not transmit at the next period. Fang *et al.* propose a new method for unsaturated stations. In this method, stations update their schedules not to forget the exact slots they transmit. Whenever a station has a new frame to transmit, if the exact slot it transmitted earlier is still idle, the station selects the same time slot and transmits in that slot. However, if the slot is already captured by another station, the station selects the same slot with the probability of γ (i.e. a parameter that controls the probability of choosing the same slot after a collision) or chooses one of the n_i idle slots with the probability of $(1 - \gamma)/n_i$. This procedure may also allow a station to select a slot that is actually reserved by another station but not used at that time due to the nature of unsaturated transmission. Therefore, even in the presence of the scheme proposed by the Fang *et al.*, unsaturated stations may collide if they have a new frame in their queue to send and select the same slot, which seems empty at the same time. As a result of a new collision, collision-free state is broken.

In UCFA, in order to prevent collisions of unsaturated stations and to maintain the collision-free state, unsaturated stations hold *virtual frames* if they do not have a frame in their queue after the last transmission. The only job of these virtual frames is to count down their deterministic backoff values whenever they sense the medium idle after the DIFS intervals as if they actually have a frame in their queue. When the backoff value of a virtual frame becomes zero and there is still no frame in the queue to transmit, it does not transmit any frame. Instead, it begins to re-count the deterministic backoff value until a new frame is inserted into the queue. Whenever the station has a new frame in its queue, then the station uses the same remaining backoff value of its virtual frame and count down from the place it left. By this way, unsaturated stations will also remember the exact slots they transmit. Therefore, it will be ideal to use a deterministic backoff value $V(d)$ for unsaturated stations as well.

The backoff countdown procedure of an unsaturated station is shown in Figure 4.3. Station 1 transmits its frame in slot two and initializes its virtual frame process since it has no more frame to transmit. Virtual frame deterministic value $V(f)$ is equalized to the value of $V(d)$ after the last successful transmission of the station. Whenever a new frame is inserted into the queue, then the station uses the same remaining backoff value of the virtual frame in order to transmit in its own slot.

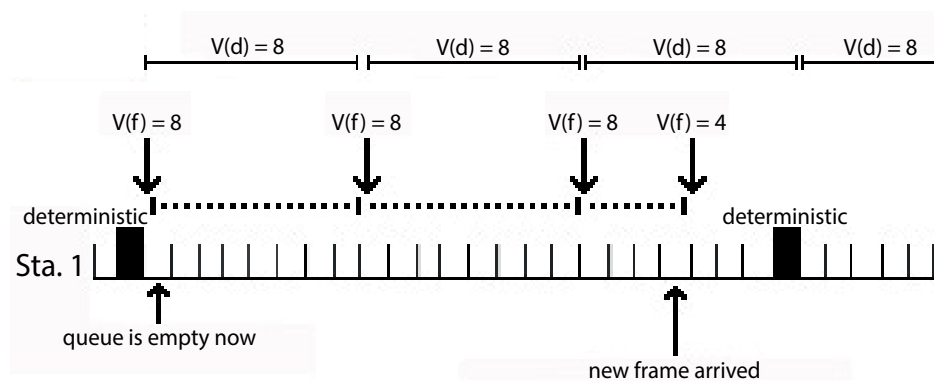


Figure 4.3: Backoff countdown procedure of an unsaturated station.

It should be noted that, in UCFA, when virtual frame backoff countdown of a station is in progress, other stations on the medium cannot sense it. Even if they cannot sense, they keep transmitting in their own time slots. However, a new station that enters the medium may capture a particular slot since it seems an empty slot.

Therefore, whenever an unsaturated station has a new frame in its queue, it first checks the previous period whether the exact slot it transmitted earlier is empty. If the slot is empty, the station transmits its frame in that slot. However, the slot might be already captured by another node. In this case, a convenient solution that detects suitable empty slots is required.

In order to increase the efficiency of slot selection, stations implemented with UCFA maintain the status (busy or empty) of each slot in their memories⁴. If a station has a new frame to transmit and its slot is already captured by another station in the previous $V(d)$, the station analyzes the status of slots in the previous $V(d)$ periods. The station marks the empty slots that are empty in all previous $V(d)$ periods as high priority slots and marks the slots that are empty in some of the previous $V(d)$ periods as *low priority slots*. Afterwards, the station primarily selects one of the *high priority slots* for its next transmission attempt. However, if there is no such slot, the station randomly selects one of the *low priority slots* in its previous $V(d)$ period. Virtual frame approach dramatically helps UCFA to prevent further collisions and maintain its collision-free state for unsaturated stations as well.

4.1. Computing the Deterministic Value $V(d)$

This section presents an adaptive deterministic backoff computation that aims at reducing both collision probability and the number of empty slots by making use of the idle and busy slots in a distributed manner at no extra cost. Online Bayesian estimator of the number of competing stations and Extended Kalman Filtering are also presented afterwards.

As it is known, APs have the information of the number of stations on the medium in the infrastructure mode. Therefore, an AP can simply broadcast the $V(d)$ value to all stations via beacon frames or via probe response frames whenever a probe request frame is received. This is the easiest way to compute the $V(d)$ value. However, this procedure requires an implementation in the AP to enable the UCFA method.

⁴In this thesis, we used a memory that is able to hold the status of all slots in between each two consecutive beacon frames.

When UCFA is the only scheme in use, stations transmit only once in a period since the selection of the transmission slot is deterministic. Therefore, stations can compute the number of stations and assign a proper $V(d)$ value with a straightforward method by simply monitoring the number of busy/idle slots in a specific interval. However, if the network is a mix of both DCF and UCFA stations, the random binary exponential backoff structure of the standard DCF scheme makes the $V(d)$ value hard to compute by monitoring the busy/idle slots. Throughout this section, we investigate whether there is an efficient method for UCFA stations to estimate the total number of stations on the medium in order to compute a proper $V(d)$ value.

Packet losses in WLANs can be classified into three types: collisions that are the result of unfavorable traffic conditions, congestions that are the result of overloaded channel and channel errors that are the result of unfavorable channel conditions. Let us consider a medium that is not overloaded (no congestion). If P_c is the probability of collision and P_e is the probability of channel error that is assumed as independent from P_c , a transmitting frame will be successful with the probability of

$$P_{success} = \frac{\#successful_trans.}{\#total_trans.} = (1 - P_c) \times (1 - P_e). \quad (4.1)$$

Suppose that over a period of time, a station transmits F frames and S of these frames are successful, because S ACKs are received. Assume there are also O slots in which the station does not transmit, and E of these O slots are empty since there is no transmission attempt in these slots. In addition, the number of slots that is consumed by DIFS intervals in O is extracted from O because stations have to sense the status of wireless medium by DIFS interval before transmitting a frame. Let us assume that a station experiences γ times DIFS interval and a DIFS interval is equal to σ slots. Therefore, over the same period of time, the number of slots that are purely used by other stations is $O - \gamma\sigma - E$. As in [Malone *et al.*, 2007], we can represent the joint likelihood function of P_c and P_e as

$$\Lambda(P_c, P_e) = \binom{F}{S} ((1 - P_c)(1 - P_e))^S \times (1 - (1 - P_c)(1 - P_e))^{F-S} \binom{O - \gamma\sigma}{E} (1 - P_c)^E (P_c)^{O - \gamma\sigma - E} \quad (4.2)$$

Using the Maximum-likelihood operation with some algebra, we can get the P_c and P_e estimated values as

$$\hat{P}_c = \frac{O - \gamma\sigma - E}{O - \gamma\sigma}, \quad \hat{P}_e = 1 - \frac{S/F}{1 - P_c} \quad (4.3)$$

Equation 3-a (P_c) computes roughly the ratio of busy slots to the sum of empty and busy slots. Hence, P_c is also an indicator of the channel utilization rate (busy probability P_b). Since an increase or a decrease at the number of stations will also increase or decrease the busy probability respectively, P_b is directly related to the number of stations on the medium. Let x_t be the number of competing stations on the medium and q be the probability that a station transmits in a given slot. Then, the probability of at least one station's transmission in a given slot, P_{tr} , is

$$P_{tr} = 1 - (1 - q)^{x_t}. \quad (4.4)$$

Bianchi *et al.* analyze the relation of probability of collision and the probability of at least one station's transmission in a given slot for the DCF as

$$q = \frac{2(1 - 2P_c)}{(1 - 2P_c)(CW_{\min} + 1) + P_c CW_{\min} (1 - (2P_c)^m)} \quad (4.5)$$

where CW_{\min} and m are the minimum contention window and the maximum backoff stage (we will explain how it is applied to UCFA shortly). It is also shown in

[Bianchi *et al.*, 2003] that, when the stations have saturated traffic, i.e., they always have a frame to transmit, then the number of competing stations x_t can be calculated as a function of the collision probability P_c as

$$x_t = f(P_c) \stackrel{\Delta}{=} 1 + \frac{\log(1 - P_c)}{\log\left(1 - \frac{2(1 - 2P_c)}{(1 - 2P_c)(CW_{\min} + 1) + P_c CW_{\min} (1 - (2P_c)^m)}\right)} \quad (4.6)$$

Since we already have the estimation of the collision probability P_c , we can also compute the estimated number of stations x_t on the medium (if stations are implemented with the standard DCF) by the equation above. Figure 4.4 shows the relation between P_c and the number of stations in a DCF-based network. Since UCFA stations do not have CW_{\min} and m values, this equation cannot be directly applied to UCFA stations. Instead, as explained below in detail, we make use of Figure 4.4 plotted by the equation above to compute a proper $V(d)$ for UCFA stations in the coexistence situation.

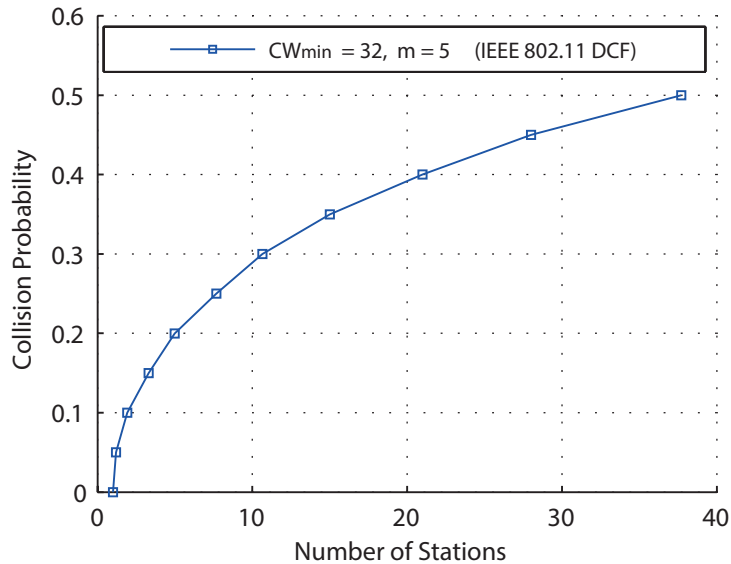


Figure 4.4: Collision probability vs. Number of stations.

As mentioned throughout the thesis, UCFA is a special case of the DCF. In the DCF, stations select a random backoff value according to their *Contention Windows* (CWs). After a collision, collided DCF stations double their *CWs* and select a new

random backoff value according to their new CWs . In UCFA, stations attempt to transmit in the same time slot in consecutive backoff periods as in TDMA. In case of a collision, if the collided station was in the deterministic state, it does not change its $V(d)$ and attempts to transmit in its time slot. Thus, UCFA aims to minimize the number of empty slots and maximize its throughput. In line with this explanation and as also shown in the simulation section (see Figure 7.6 and Figure 7.7), an UCFA station occupies more slots than a DCF station in any case of the coexistence situation.

Let us consider two different networks. One is composed of DCF stations and the other one is composed of both DCF and UCFA stations. If the numbers of stations in these networks are equal (e.g. 8 DCF stations and 4+4 or 2+6 DCF+UCFA stations), the number of busy slots in between two consecutive beacon frames will be higher in the coexistence situation compared to the standard DCF situation. Therefore, estimated P_b value (i.e. equal to P_c), which is the ratio of busy slots to the sum of empty and busy slots, will be higher in the coexistence situation as well. This observation indicates that the computed P_b value in the coexistence situation will not be lower than the P_b value computed in the DCF situation since the numbers of stations on these mediums are equal both for the DCF and the coexistence situation. This observation is valid for any number of stations on the medium. Therefore, the curve in Figure 4.4 (the curve of the DCF stations) can be used as a lower-bound of the number of stations in the coexistence situation. In other words, if we compute the number of stations x_t by matching the P_b value (computed in the coexistence situation) with the curve in Figure 4.4, computed x_t in the coexistence situation will always be higher than the x_t computed in the DCF situation since the curve in Figure 4.4 is used as a lower-bound.

In UCFA, stations hold the P_b values corresponding to each number of stations in the Figure 4.4 in their memories. Thus, UCFA stations can estimate the number of stations x_t by computing the P_b value through the equation 3 and then making use of the curve in Figure 4.4. As explained above, this process leads to an x_t that is higher than the actual value. However, computing the P_b and hence the x_t higher than the actual value is not bad for the coexistence situation. In order to have sufficient slots for the existing stations and to let new stations enter the medium anytime without causing collisions, $V(d)$ value must be higher than the actual value of x_t . Therefore, in UCFA, the $V(d)$ is assigned as a function of $V_{dconvergence}$, which converges the sum of

x_t and the value of possible new entrants x_n (i.e., 1 in the simulations) to the closest higher value of five predetermined $V(d)$ periods, i.e., 4, 8, 16, 32 and 64, respectively.

$$V(d) = V_{d_{convergence}}(x_t + x_n) \quad (4.7)$$

For example, if the number of competing stations, x_t , is estimated as 12, then the sum of the x_t (12) and x_n (1) becomes 13 and the $V(d)$ is assigned to 16. In other words, in case the x_t value is estimated in between 8 and 15, the $V(d)$ is assigned to 16. This procedure allows stations to have the same $V(d)$ value even if the estimation method in each node computes the number of contending stations a bit different. In this context, since the computed x_t will be higher than the actual value in the coexistence situation, $V(d)$ will be higher than the actual x_t as well. This will allow UCFA and DCF stations to have sufficient slots to transmit.

In order to allow stations to provide a run-time adaptive estimation of x_t in a distributed way, we use an ARMA (Auto Regressive Moving Average) filter given in [Bianchi *et al.*, 2003]. In this filter, the number of competing stations is estimated from $\hat{p}(t)$ (i.e., an ARMA smoothing of the collision probability p) by making use of the number of busy/idle slots and the nonlinear curve shown in Figure 4.4. However, due to the slope of the curve shown in Figure 4.4, which plots $x_t = f(P_c)$ as given by the above equation, accuracy of this estimation degrades as the number of stations increases. The slope increases in parallel with the increase in the number of stations. Therefore, error ratio of the collision probability estimation increases in the evaluation of high number of contending stations.

In order to achieve a better estimation of the number of stations on the medium, authors in [Bianchi *et al.*, 2003] also propose an Extended Kalman Filtering (EKF) technique that provides a state and a measurement model. The state of the number of stations x_t is assumed to fluctuate as

$$x_t = x_{t-1} + w_t \quad (4.8)$$

where w_t is the state noise with variance Q_t and the probability of collision is

$$p_t = Bh(x_t) + o_t \quad (4.9)$$

where B is the total number of slots observed, h is the inverse function of the equation 6, and o_t is the observation noise with variance V_t

$$V_t = Bh(x_t)(1 - h(x_t)) \quad (4.10)$$

In order to fit the EKF assumptions, all variables are used as continuous variables. Q_t and V_t are assumed to be known. Additionally, the EKF approach provides the following update procedure

$$\hat{x}_t = \hat{x}_{t-1} + K_t z_t \quad (4.11)$$

where z_t is the innovation given by the t^{th} measure. The Kalman gain K_t is specified as

$$K_t = \frac{(P_{t-1} + Q_t)s_t}{(P_{t-1} + Q_t)s_t^2 + V_t} \quad (4.12)$$

where s_t is the sensitivity of the measurement

$$s_t = B \left. \frac{\partial h(x)}{\partial x} \right|_{x=\hat{x}_{t-1}} \quad (4.13)$$

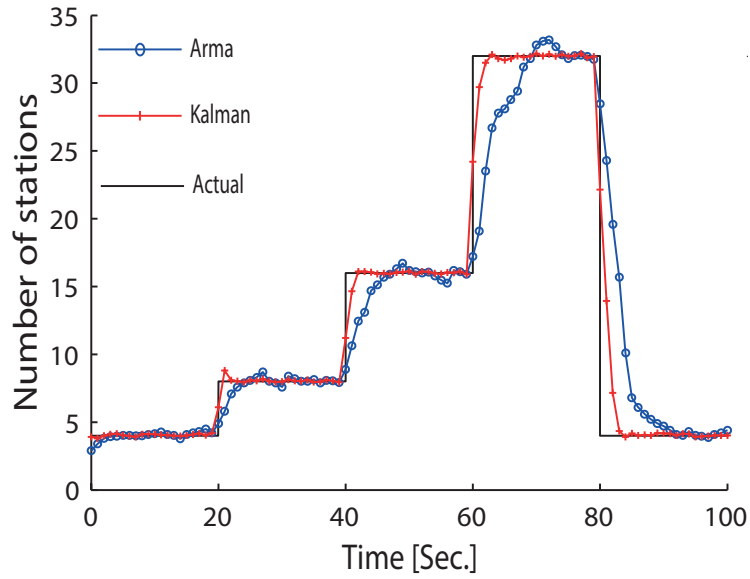
It should be noted that there is a tradeoff between the estimation accuracy and the tracking ability in the Kalman Filter estimation. Therefore, the variance of the state noise, Q_t , needs to be adjusted. High values of Q let the estimation react to the state changes rapidly. However, it implies a reduced accuracy in the estimation. In

contrast, low values of Q achieve a better estimation in stationary condition, but a very slow transition phase when sudden state variations occur [Bianchi *et al.*, 2003].

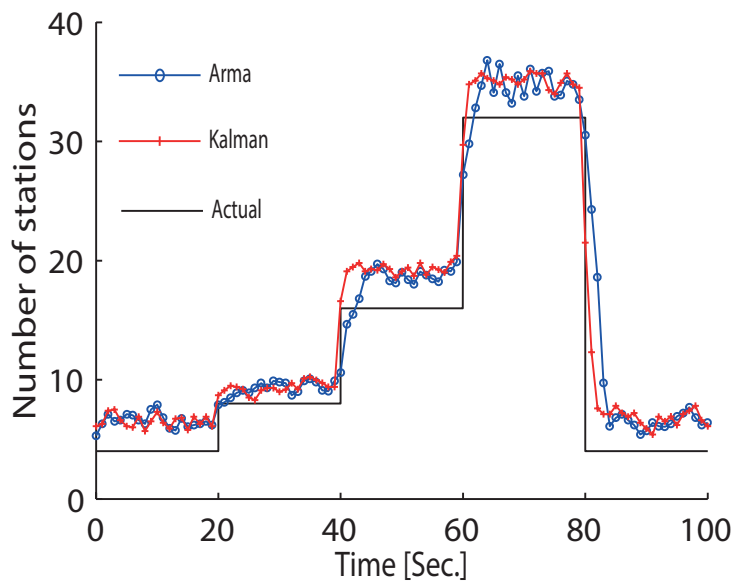
In this context, the number of competing terminals is first assumed to be constant and Q_t is assigned to 0 (zero) in UCFA. Since stations implemented with UCFA have the status (busy/idle) of each slot in $V(d)$ periods, whenever the status of a specific number of slots⁵ in between two consecutive beacon frames changes (due to an entrance/termination of a station or nature of an unsaturated station's transmission), then stations assign a large Q_t , i.e., 5 to react to the state changes rapidly.

In order to evaluate the accuracy of the Bayesian and the Extended Kalman Filter estimations, Bianchi *et al.* performed tests over a wide range of network conditions (both for saturated, unsaturated and error-prone networks) using simulators. Their simulation results show the robustness of these estimators. Due to the scarcity of space, we do not see a need to repeat the evaluation of these estimators in this thesis. Detailed analysis and evaluation of these estimations can be found in [Bianchi *et al.*, 2003]. However, in order to evaluate the accuracy of this filters in a mixed scenario where UCFA and DCF stations run together, we simulated a scenario in which the number of stations in the network is set to be 4 at the first 20 seconds, then changed to 8, 16, 32 and 4 at the 20th, 40th, 60th and 80th seconds, respectively. Performance of the proposed ARMA and Kalman filters for the standard DCF and the mixed scenario (half of the stations are implemented with DCF, and the other half with UCFA) can be seen in Figure 4.5.

⁵In this thesis, we assign Q_t as 5 whenever the status of more than 20% of the total slots in between two beacons changes.



a)



b)

Figure 4.5: Accuracy of the ARMA and Kalman filters for the Standard DCF and the DCF-UCFA mixed scenario. a) DCF stations, b) DCF-UCFA stations in half shares.

As is shown in Figure 4.5-b, both ARMA and KALMAN filters are not very accurate at the estimation of x_t in a mixed scenario. Estimated values are mainly higher than the actual number of stations. It is because specific parameters of UCFA stations are not assigned as the DCF stations ($CW_{min}=32$, and $m=5$). However, estimating the x_t higher than the actual value is better for both UCFA and DCF

stations in a mixed scenario⁶. It is because assigning a bigger $V(d)$ value ($V(d) = V_{\text{convergence}}(x_t + x_n)$) allows to have sufficient number of slots both for UCFA and DCF stations. It also improves the fairness between the DCF and UCFA stations. Therefore, we use the EKF estimation proposed in [Bianchi *et al.*, 2003] to compute the number of stations x_t when the network is a mix of both DCF and UCFA stations.

Consequently, our extensive simulations shows that throughput improvement and sufficient slot allocation for both UCFA and DCF stations can be achieved if we use the EKF filter to compute the number of station x_t when the network is a mix of both DCF and UCFA stations. By this method, if there are many UCFA stations and only few DCF stations, estimated x_t will be higher than the actual value and hence, $V(d)$ will be assigned higher as well. Therefore, DCF stations will have sufficient slots to access the medium. If there are many DCF stations and only few UCFA stations, then the estimated x_t will be closer to the actual value and hence, UCFA stations will be able to increase their throughputs assigning a realistic $V(d)$.

It should be noted that the derivation of the $V(d)$ is obtained in the assumption of no hidden nodes. In the presence of hidden node(s), stations that are aware of the hidden node(s) may compute the $V(d)$ different than the stations that are not aware of the hidden node(s). As a result, overall throughput may decrease (see Figure 13 in the simulations section).

Additionally, in CSMA/CA – TDMA based hybrid schemes (ZC, L-BEB, etc.), if the cycle length is less than the number of users in the network, transport layer congestion avoidance mechanism may help TCP stations to avoid congestion by reducing the rate of sending frames. However, even in the presence of congestion avoidance mechanism, there will be some stations attempting to transmit in a slot that is reserved by another station since they do not have a slot reserved by themselves (the cycle length is assigned less than the number of users in the network). In this case, if a station that has a reserved slot and a station that has no reserved slot transmit at the same time, collisions occur. Therefore, congestion

⁶As an example, assume there are 7 UCFA and 7 DCF stations on the medium. In this case, it first seems that $V(d)$ value for UCFA stations may be assigned to 16 since there are less than 16 stations on the medium. We first run the simulation as there are 7 UCFA and 7 DCF stations and assigned the first $V(d)$ value of UCFA stations to 16. During the first seconds of the simulation, we observed that most of the UCFA stations doubled their $V(d)$ s (see the pseudo-code of Algorithm 2) due to the collisions caused by the insufficient number of slots. Upon receiving the next beacon frame, we computed the number of stations x_t on the medium using the EKF estimation method. x_t (14) is computed as 17,6 which is not very accurate. It is because UCFA stations try to capture the slots in every $V(d)$ slot in contrast to the standard DCF stations that double their CWs with each unsuccessful transmission attempt. Since $V(d)$ is assigned through the sum of $x_t + x_n$, then $V(d)$ would be equal to 32 in case the x_t (14) is estimated as 17,6 by the EKF filter ($V(d) = V_{\text{convergence}}(17,6 + 1) = 32$). We also run the simulation as $V(d)$ is fixed to 16 and 64, respectively for all UCFA stations but observed much less throughput than the setup when the $V(d)$ is assigned to 32.

avoidance may solve the congestion but not the collision. However, in UCFA, since stations will adaptively double their $V(d)$ s when the estimated number of users exceeds the previous $V(d)$, the probability of congestion and collision will be reduced (congestion and collision will be prevented if there are only UCFA stations in the network) by supplying enough number of slots in one deterministic period for stations.

4.2. Entrance of New Nodes and Maintaining the Collision-Free State

Whenever a new station enters the medium, the standard CSMA/CA and other hybrid schemes randomly select a new backoff value for the station and attempt to transmit that frame according to its new backoff value. Since the new station does not know the deterministic backoff value, it cannot compute its period and cannot find the empty slots it needs to transmit its frame. As a result, random backoff selection of this new station may cause collision(s). After a collision, more than one station has to retransmit their frames and they have to select new random backoff values for their frames. These new backoff values of stations may cause new collisions in the network and may initiate a chain reaction that brings the system to its transitory CSMA-like operation.

Instead of random backoff selection, a new station associated with the network waits to receive the next beacon frame before its first transmission attempt in UCFA. Meanwhile, it maintains the status (busy or empty) of each slot in its memory. Upon receiving the new beacon, the station computes the x_t and the deterministic backoff value $V(d)$, respectively as it is covered in the previous section. Then, the backoff selection procedure is similar to the slot selection procedure of a virtual frame. Instead of random selection of the backoff value, the station analyzes the status of slots in the previous $V(d)$ periods. The station marks the empty slots that are empty in all previous $V(d)$ periods as high priority slots and marks the slots that are empty in some of the previous $V(d)$ periods as *low priority slots*. Afterwards, the station primarily selects one of the *high priority slots* for its next transmission attempt. However, if there is no such slot, the station randomly selects one of the *low priority slots* in its previous $V(d)$ period.

In this way, if the network is composed of the saturated stations, a new station that enters the medium selects one of the empty slots, transmit in that slot and maintain its collision-free state. If the network has both saturated and unsaturated stations, despite of the new station and unsaturated nature of some stations, the probability of maintaining the collision-free state for UCFA is much higher than all the existing standards and hybrid approaches. It is because UCFA differentiates the real empty slots and the slots that are not used due to the unsaturated nature of some stations. Here, waiting cost for the new station before transmitting its first frame is the only cost that it waits for the new beacon frame after the association to the network. In other words, the total waiting cost for the new station is about 100ms. Since this interval can also be counted as the connection time cost before sending the first frame for the new station, it will not negatively affect the QoS of the station itself, ongoing multimedia stations or data stations. Besides, since the collision-free state will be going on, it will improve the performance of both data traffic by improving the throughput and multimedia traffic by preventing the possible frame losses, delay and delay variation.

The procedure for this new approach is shown in Figure 4.6. As is shown in the figure, upon receiving a new beacon frame, the new station calculates the $V(d)$ through x_i , analyzes the previous $V(d)$ periods and then randomly selects one of the empty slots in the previous $V(d)$ period. The new station transmits its first frame whenever its backoff value becomes zero. After the successful transmission, the new station also acquires a particular slot and keeps transmitting in the same slot for its next transmissions.

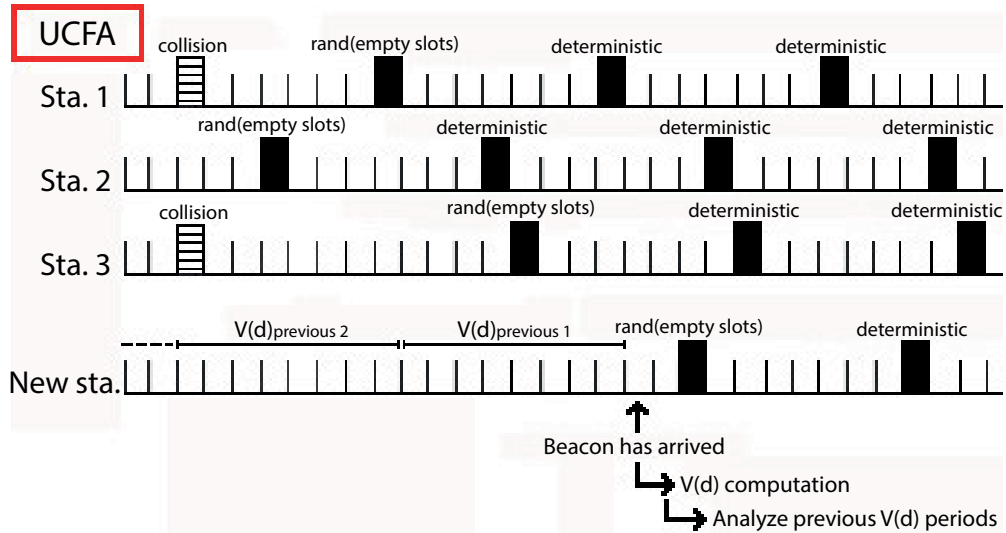


Figure 4.6: Procedure of a new entrant in UCFA.

Another feature that distinguishes the UCFA from the other existing hybrid approaches is the reaction after a collision or a frame error. While hybrid approaches randomly selects a backoff value for the stations that had a collision or frame error (ZC randomly selects one of the empty slots in the previous deterministic period), UCFA reacts to the collision or frame error in two different ways. If the station that had a collision or frame error was in the *deterministic-state* (successfully transmitted its previous frame and used the deterministic backoff value $V(d)$ for its last transmission), it keeps using the same $V(d)$ for its backoff value and expects the other station(s) (if there was a collision) to change its backoff value. However, if the station that had a collision or frame error was not in the *deterministic-state*, then it selects one of the empty slots in the previous period randomly for its next transmission attempt as in ZC [Lee *et al.*, 2008]. Since only one station at a time can get the $V(d)$ for a particular slot, obviously, the other collided station(s) will have to select one of the empty slots. In this way, in case of a collision or a frame error, only the station(s) that does not have the $V(d)$ has to select a new backoff value. This approach avoids unnecessary new collisions that may initiate a chain reaction that brings the system to its transitory CSMA-like operation.

Furthermore, due to bad channel condition or excessive usage of the network, if the consecutive failed transmission count of a station, m_t , exceeds a pre-determined threshold (m_t is assigned to 3 in the simulations), instead of random backoff selection as in the standard CSMA/CA or existing hybrid approaches, UCFA multiplies the

$V(d)$ by two for that station. This procedure provides higher backoff values for stations under bad channel condition or excessive usage of the network (however, this procedure may also result in short term unfairness). Once the new beacon frame is received by the stations, stations re-compute the deterministic backoff period and use that value afterwards.

As a worst case scenario, let us assume that a node(s) miscalculates the $V(d)$ and hence, has a lower deterministic backoff value than the rest of the stations on the medium. Even in this case, since the predetermined deterministic backoff values (4, 8, 16, 32 and 64) are multiples of each other, collisions may not occur⁷. However, if collisions occur, as shown in pseudo-code of the UCFA (Algorithm 2), the node(s) doubles its $V(d)$ whenever the retransmission count of a frame t_a exceeds the number of maximum transmission attempts m_t (i.e., is assigned to 3 in the simulations). This procedure allows the node(s) to avoid further collisions and lets the system enter a collision-free state again. Additionally, if a node(s) miscalculates the $V(d)$ and hence, has a higher deterministic backoff value than the rest of nodes, no collisions occur, collision-free state continues and the node transmits as much as half of the other nodes until a new $V(d)$ is assigned.

Algorithm 2 in the Appendix shows the pseudo-code of the UCFA that is implemented in each of the contending stations in a distributed manner. In this regard, t_a is the number of transmission attempts, m_t is the maximum number of transmission attempts, x_t is the number of contending stations, x_n is the number of possible new entrants, $V(d)$ is the deterministic backoff value, *backoff* is the backoff counter and V_f is the virtual frame counter, respectively.

4.3. Adaptation to the EDCA: Traffic Prioritization

EDCA offers differentiated service by providing different waiting times for different traffic priority levels. Three metrics have to be examined for the traffic differentiation, namely, the Arbitration Inter-Frame Space (*AIFS*), Transmit opportunity (*TXOP*) and Contention Windows (CW_{min} and CW_{max}). Since *TXOP* does not have any impact in the contention procedure, application of *TXOP* is a valid

⁷ e.g. $V(d)$ is 16, but a node assigns its $V(d)$ as 8. Let's say the transmission slot of the node is slot 3. If both slot 3 and slot 11 ($3+V(d)$) are empty, then no collision occurs and the node transmits two times more than the rest of the nodes until a new $V(d)$ is assigned.

option for traffic differentiation in UCFA. However, utilization of *AIFS* violates the assumption that all the stations decrement their backoff simultaneously. Therefore, it is impractical in UCFA. Ultimately, traffic differentiation can also be achieved by choosing different deterministic backoff values for high and low-priority levels (CW_{high} and CW_{low} , respectively). If values of the contention windows are selected as powers of two (CW_{high} is an integer divisor of CW_{low}), the adaptation of different deterministic values allows stations to maintain the collision-free state. Therefore, it is also a valid option for UCFA.

4.4. Fairness and Additional Observations

Since the deterministic backoff value in UCFA is computed adaptively using the empty and busy slots in the network, $V(d)$ will be higher than the number of stations in the network even in the coexistence situation of UCFA and the DCF. Therefore, during a deterministic period, medium will have enough slots for both UCFA and DCF stations. In this case, stations using UCFA will virtually try to capture some of the slots and stations implemented by DCF will randomly select some of slots. Collisions will be mainly in between UCFA and DCF stations since UCFA stations rarely collide each other due to the procedure explained throughout the thesis. Thus, collision probability of the coexistence scenario will be lower than the collision probability of the standard DCF scenario.

It is worthy of note that if UCFA is the only scheme in use, stations on the medium have approximately the same amount of throughput since each station periodically transmit in every $V(d)$ slot. Therefore, UCFA provides fairness. Additionally, if UCFA is implemented as traffic prioritization in mind (as in EDCA), then nodes that have the same type of traffic will have approximately the same amount of throughput as in EDCA. However, if the medium is composed of both DCF (or EDCA) and UCFA stations, then an evaluation is required to determine whether nodes are receiving a fair share of the system resources. Therefore, fairness index defined by Jain *et al.* is implemented in the simulation section to evaluate the degree of fairness for the standard DCF, UCFA and a mixed scenario. This fairness index is defined as,

$$J(x_1, x_2, \dots, x_n) = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n x_i^2} \quad (4.14)$$

where n is the number of flows and x_i is the throughput of the i^{th} flow. Results ranges from $1/n$ (worst-case) to 1 (best-case), and it is maximum when users receive the same allocation.

It should also be noted that when the standard CSMA/CA or UCFA (or another related hybrid method) is in use, the hidden node problem occurs when a node is visible from a wireless access point (AP), but not from other nodes communicating with the AP. Since the node(s) cannot sense the carrier, CSMA techniques do not work. Nodes may transmit at the same time and collisions occur. The “popular” solution to the hidden node problem appears to be the RTS/CTS frames, but as it is shown in the simulation section and other related works [Haitao *et al.*, 2006], this solution may not work that well especially when there are few number of hidden nodes on the medium. Other solutions, such as increasing the transmitting power of nodes, using omni-directional antennas, removing obstacles, moving the node(s), using protocol enhancement software, using antenna diversity and etc. are all applicable for the CSMA/CA, UCFA and other hybrid approaches.

5. BEACON-BASED COLLISION-FREE CHANNEL ACCESS

As it is mentioned in earlier sections, CSMA/CA is fairly simple to implement but collisions may cause inefficient use of the medium. TDMA on the other hand, eliminates the possibility of collisions. However, set-up is required and a central unit has to maintain detailed knowledge of each station in TDMA, which requires additional data exchange. Therefore, borrowing the idea of reservation from TDMA and applying it in wireless CSMA/CA networks can enhance the performance of DCF and EDCA.

The proposed scheme that we call Beacon-based Collision-free Channel Access (BCCA), assumes that all stations on the medium are bounded by a deterministic backoff period⁸; $V(d)$. Certain number of time slots constitutes the deterministic backoff period. If a station successfully transmits its frame in a time slot, then the backoff value of the station is assigned as the deterministic backoff period, $V(d)$. This procedure lets the station use the same time slot in consecutive backoff periods as in TDMA. Since selection of the transmission slot is deterministic over a period, as soon as all the stations successfully transmit and capture their slots, each of the stations periodically transmits in every $V(d)$ slot. Hence, collisions disappear, and stations orderly transmit in a TDMA fashion.

In BCCA, each station maintains and updates the set of empty slots of the last $V(d)$ period in a distributed manner. As it is known, a slot is empty if no transmission occurs in that slot and all the contending stations can detect that slot as an empty slot. In BCCA, a station ready to transmit its first frame after associating with the AP, waits for one deterministic backoff period; $V(d)$, i.e. 16 slots, before attempting to transmit. If there are empty slots between these slots, then the station randomly and uniformly selects one of the empty slots of the last $V(d)$ period and transmits in that slot. As long as the transmission is successful, the station remembers that slot as being reserved by itself and uses the same deterministic backoff value $V(d)$ for its next transmission attempts.

⁸ $V(d)$ is chosen 16 for the first transmission attempt throughout the simulations since the CW_{\min} is set between 0 – 31 in CSMA/CA DCF.

In order to decrease the number of idle slots and increase the channel efficiency, each station initializes its *approaching process* upon receiving each beacon frame in BCCA. This process allows each station to check whether there are empty slot(s) in between the slot of the station itself and the slot of a station that has a lower backoff value. If there are empty slots between these two stations, upon receiving a beacon frame, the station that has a higher backoff value decreases its backoff value, bypassing empty slots, and becomes the neighbor station to the nearest station.

Procedure of the deterministic backoff selection in BCCA and in CSMA/CA is shown in Figure 5.1 and Figure 5.2, respectively. Although actual duration of empty, successful and collision slots is different, they are all represented equally in the figures for simplicity. In the figures, five saturated stations are contending for the channel. Black and striped boxes represent successful transmissions and collisions, respectively. While BCCA selects a deterministic (det.) backoff value after a successful transmission and a random value, *rand(e.s)*, between empty slots in case of a collision, CSMA/CA always chooses a random backoff value. In BCCA, after all stations successfully transmit, they acquire a particular slot and keep transmitting in that order. Therefore, the system enters its *collision-free state*⁹. However, collisions may occur anytime in CSMA/CA due to the standard random binary exponential backoff procedure.

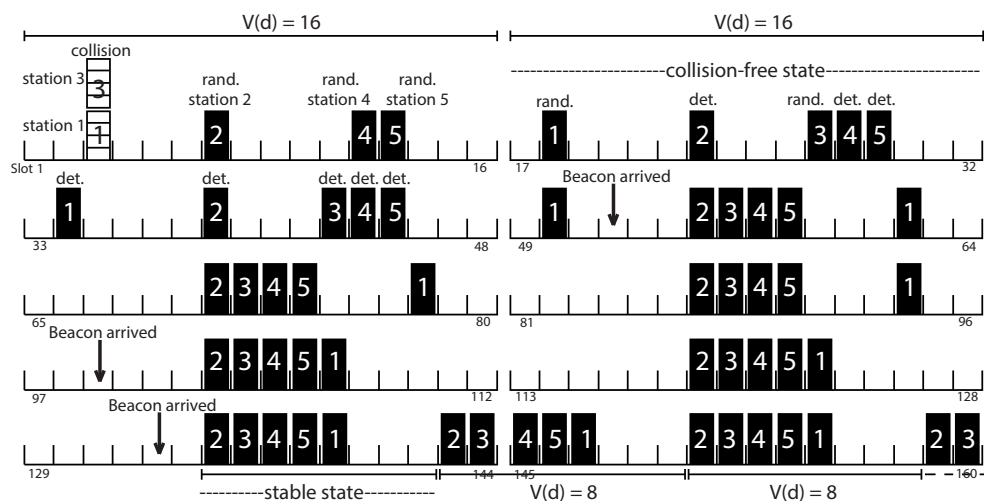


Figure 5.1: BCCA deterministic backoff selection procedure.

⁹We say that the network has reached to the *collision-free state* when all stations in the network have reserved a different slot in a backoff period.

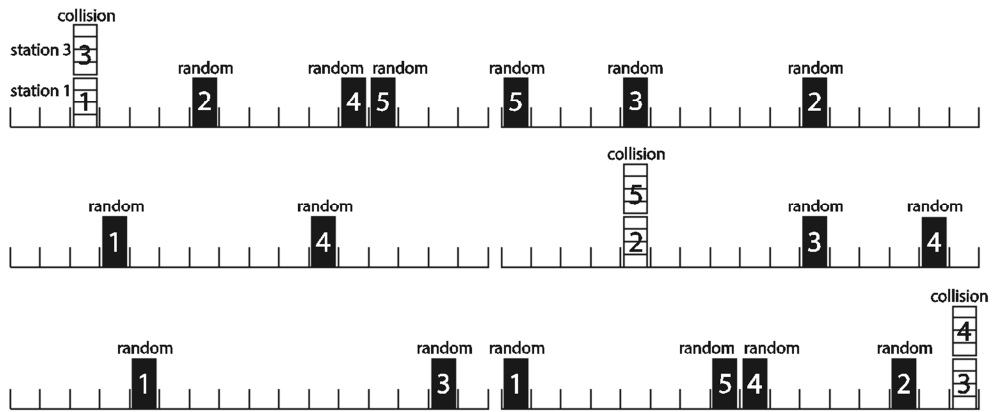


Figure 5.2: Binary Exponential Backoff procedure in CSMA/CA.

As it is demonstrated in Figure 5.1, while station 1 and station 3 collide each other, station 2, 3 and 4 successfully transmit. Since these stations successfully transmit, they enter the deterministic phase and use the same backoff value by assigning it as the value of $V(d)$ for their next transmission attempts. However, station 1 and 3 randomly select one of the empty slots of the previous period. In this case, the empty slots are the slot 1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 14, 15, 16. At the next period, station 1 and 3 randomly select the slot 2 and 11, respectively. Since all stations on the medium transmit successfully in this period, the medium enters the *collision-free state* and all stations keep transmitting in their own particular slot until a beacon frame is received. As it is seen in Figure 5.1, the first beacon frame is received in the slot 4 of the fourth period. Upon receiving the beacon frame, all stations initialize their *approaching process* in a distributed manner. Since the station 2 is the first station to transmit after the beacon frame (no station to approach), it does not change its position. However, station 3, 4, and 5 will change their slots since station 2 has a lower backoff value and there are empty slots between station 2 and them. Station 1 will also change its position since station 5 has a lower backoff and there are empty slots between station 5 and station 1.

Let us take a look at the reaction of the station 5. Station 5 captures the 13th slot in each $V(d)$ period until a beacon frame is received. After the *collision-free state*, station 3, 4 and 5 line up side-by-side. Upon receiving a beacon, since the station 2 in the 7th slot has captured a lower slot than station 5 and there are empty slots in between, new backoff value of station 5 becomes,

$$backoff_{sta\ 5} = backoff_{approached\ sta.} + x_{neighbor} + 1 \quad (5.1)$$

where $backoff_{approached\ sta.}$ is the remaining backoff value of approached station (station 2) and $x_{neighbor}$ is the number of neighbor stations that have lower backoffs (station 3 and 4). Therefore, after the beacon frame, $backoff_{sta5} = 3 + 2 + 1 = 6$. Similarly, backoff values of the station 3 and 4 become 4 and 5, respectively. Thus, stations 2, 3, 4 and 5 line up side-by-side and become neighbor stations, bypassing empty slots. Additionally, new backoff value of the station 1 is, $backoff_{sta1} = 9 + 0 + 1 = 10$. In this case, the approached station is the station 5 and its backoff is 9. Note that the new backoff value of station 5 is 5 but since the station 1 initializes its *approaching process* immediately as the other stations do after receiving the beacon frame, station 1 assign its new backoff using the previous backoff value of the station 5.

Using this procedure, upon receiving a certain number of beacon frames, all stations line up side-by-side and become neighbor stations. If all stations in a $V(d)$ period line up side-by-side, after receiving a new beacon, stations reassign their deterministic backoff value. This new value is used as the new $V(d)$ by all stations and it is the sum of the number of stations and the *number of new stations*¹⁰ possible to enter the medium.

$$V(d)_{new} = x_t + x_e \quad (5.2)$$

5.1. Expected Convergence Time

In this section, we show that BCCA converges to the *collision-free state* in finite time as long as the $V(d)$ is bigger than x_t . We then, analyze the expected number of cycles and the number of beacon frames before network converges to its *stable state*¹¹.

¹⁰In simulations, x_e is set to the value that makes $V(d)_{new}$ multiples of 4, such as if $x_t = 3$, $x_e = 1$ or if $x_t = 5$, $x_e = 3$, and so on.

¹¹We say that network has reached to its *stable-state* when $V(d)$ is reassigned after all stations capture the slots side-by-side.

In BCCA, each station attempts to transmit only once in $V(d)$ consecutive slots and the number of stations x_t is less than the deterministic backoff period $V(d)$. It is because, stations implemented with BCCA always observe the slots of the last deterministic backoff period, and as it will be explained in the next section, in case of all slots are busy (reserved) in a period, stations reassign (enlarge) their deterministic backoff period.

Now, assume that x_n is the number of stations that have particular slots after the n -th cycle of $V(d)$. In this case, $\{x_n, n \geq 0\}$ is a Markov chain. As it is proved in [Lee *et al.*, 2008], the probability that exactly k stations among x_t select a slot that no other stations select for $0 \leq k \leq x_t$ is

$$p_{v(d), x_t}(k) = \sum_{j=k}^{x_t} (-1)^{j-k} \binom{x_t}{j} \binom{j}{k} \frac{v(d)!(v(d)-j)^{x_t-j}}{(v(d)-j)!v(d)^{x_t}} \quad (5.3)$$

Since, when $x_n=m$, there are $x_t - m$ stations left with unreserved slots and $V(d) - m$ remaining slots to choose from

$$p_{m, m+k} := P[x_{n+1} = m + k | x_n = m] = p_{x_t-m, v(d)-m}(k). \quad (5.4)$$

Next, we study the mean time until x_n reaches the state x_t . Let $L = \min\{n \geq 0 | x_n = x_t\}$. In order to calculate $E[L | x_0 = 0]$, let's define $\beta(i) = E[L | x_0 = i]$. Then, as in [Lee *et al.*, 2008],

$$\beta(i) = \begin{cases} 1 + \sum_{j=i}^{x_t} p_{i,j} \beta(j), & 0 \leq i \leq x_t - 1 \\ 0, & i = x_t \end{cases} \quad (5.5)$$

Solving these equations yields $E[L | x_0 = 0] = \beta(0)$. If the cycle n has S_n successful transmissions, I_n idle slots, and $C_n = V(d) - S_n - I_n$ slots with collisions, the length of the cycle n , T_n , is

$$T_n = t_s S_n + t_i I_n + t_c C_n + t_g V(d) = (t_g + t_c)V(d) + (t_i - t_c)I_n + (t_s - t_c)S_n \quad (5.6)$$

where t_s , t_i and t_c are the durations of a successful transmission, an idle slot, and a collision, respectively. Finally, t_g is the inter-slot gap. These duration parameters are calculated based on the standard IEEE 802.11 MAC/PHY specification, where

$$t_s = 2 \times \left(T_{PLCP_{preamble}} + T_{PLCP_{header}} \right) + T_{MPDU} + T_{SIFS} + T_{ACK} \quad (5.7)$$

$$t_i = T_{SLOT} \quad (5.8)$$

$$t_c = T_{PLCP_{preamble}} + T_{PLCP_{header}} + T_{MPDU} + T_{EIFS} \quad (5.9)$$

$$t_g = 0 \quad (5.10)$$

Given the stopping period L , the total convergence time to *collision-free* state, T_{cf} is

$$T_{cf} = \sum_{n=1}^L T_n = \sum_{n=1}^L t_s S_n + t_i I_n + t_c C_n + t_g V(d) \quad (5.11)$$

In the equation (6), $(t_s - t_c)$ can be either positive or negative. If $(t_s - t_c)$ is positive, it is shown in [Lee *et al.*, 2008] that

$$\begin{aligned} E[T_n|L] &\leq (t_g + t_c)V(d) + (t_s - t_c)x_t + (t_i - t_c)(V(d) - x_t) \\ &= (t_g + t_i)V(d) + (t_s - t_i)x_t \end{aligned} \quad (5.12)$$

If $(t_s - t_c)$ is negative,

$$\begin{aligned}
E[T_n|L] &\leq (t_g + t_c)V(d) + (t_s - t_c) \times 0 + (t_i - t_c)(V(d) - x_t) \quad (5.13) \\
&= (t_g + t_i)V(d) + (t_c - t_i)x_t
\end{aligned}$$

Therefore, upper bound of the expected convergence time is

$$E[T] \leq \{(t_g + t_i)V(d) + (\max(t_s, t_c) - t_i)x_t\} E[L]. \quad (5.14)$$

Now, we calculate the expected convergence time to the *stable-state* for the worst-case scenario. To this end, we assume that stations take T_{cf} times to converge to the *collision-free state*. After the *collision-free state*, the BCCA allows each station to initialize its *approaching process* upon receiving each beacon frame. In the best case scenario, all stations capture the slots that are side-by-side (no idle slots between reserved slots) and it takes only one beacon to converge to the *stable-state*. In the worst case scenario, there would be idle slot(s) between each station. Then, upon receiving each beacon frame, stations approach to their neighbor stations. Finally upon receiving sufficient number of beacon frames, all stations capture the slots side-by-side, reassign their deterministic backoff value $V(d)$ and converge to the *stable-state*. Therefore, the total convergence time to the *stable-state*, T_{ss} is

$$T_{ss} = \sum_{n=1}^L T_n + T_{approaching}. \quad (5.15)$$

Let $\Pi_i()$ be the number of stations that become a neighbor with another station bypassing empty slots after receiving the i -th beacon frame. Assume that $x_t \leq V(d)$ and the medium is not at the stable-state (there are empty slot(s) between stations). Then, upon receiving a beacon frame, for $0 \leq i \leq x_t$, we say that $\Pi_i() \geq 1$.

Proof:

Since there are empty slot(s) between stations, upon receiving a beacon frame, there will be station(s) that change their backoff values according to the equation (1).

In this case, at least one station in the medium will be a neighbor with another station and $\Pi_i()$ will be bigger than zero.

Let $f_i(x_t, V(d))$ be the number of gaps between stations after receiving the i -th beacon frame. In this case, we say that the maximum value of $f_i(x_t, V(d))$ is x_t .

$$f_i(x_t, V(d))_{max} = x_t \quad (5.16)$$

Proof:

Let us consider the deterministic backoff period as a cycle. Assume there are only two stations in the medium and there are empty slot(s) between these two stations. In this case, as it is seen in Figure 5.3-a, the number of gaps between these stations is 2 (the gaps on the left and right side of the cycle). If the number of stations, which have empty slot(s) before and after, increases, then the number of gaps also increases as shown in Figure 5.3-b. Since neighbor stations that have no empty slot(s) in between is counted as one station, in order to find the maximum number of gaps, we must assume that all stations have empty slot(s) before and after. In this case, the number of gaps is maximized and clearly, it can be x_t at maximum.

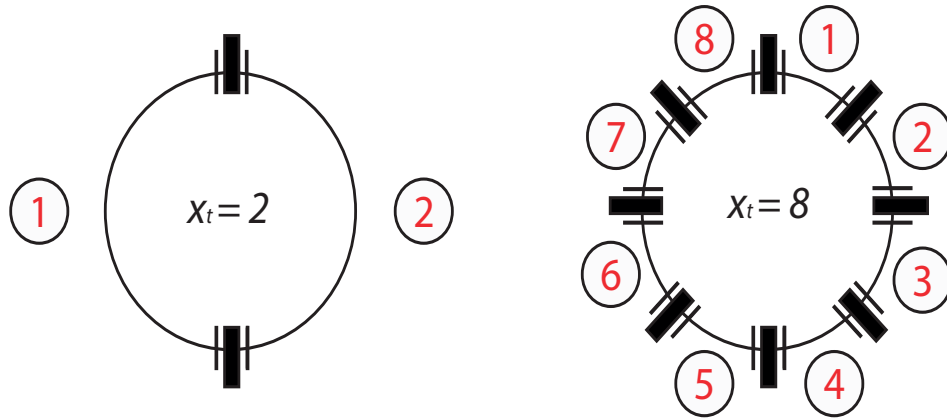


Figure 5.3: Maximum number of gaps for various numbers of stations.

Since $\Pi_i()$ pieces of stations bypass the empty slots upon receiving each beacon frame, the value of the $f_i(x_t, V(d))$ after receiving the i -th beacon frame is,

$$f_i(x_t, V(d)) = x_t - \sum_{n=1}^i \Pi_n() \quad (5.17)$$

Therefore, at the worst case scenario, even if only one station become a neighbor with another station ($\Pi_i() = 1$) after each beacon, the number of gaps between stations after receiving the $x_t - 1$ beacon frame is,

$$f_{x_t-1}(x_t, V(d)) = x_t - \sum_{n=1}^{x_t-1} \Pi_n() = x_t - (x_t - 1) = 1 \quad (5.18)$$

This means, at the worst case scenario, after $x_t - 1$ beacon frames, all stations become neighbors with each other. Upon receiving the next beacon frame, stations reassign their deterministic backoff period as, $V(d)_{new} = x_t + x_e$. Finally, the total convergence time to the *stable-state*, T_{ss} is

$$T_{ss} = \sum_{n=1}^L T_n + T_{approaching}, T_n \leq T_{approaching} \leq T_n x_t \quad (5.19)$$

5.2. Entrance of New Nodes and Maintaining the Collision-free State

In BCCA, a new station associated with the network, waits for one $V(d)$ period before its first transmission attempt. If there are empty slot(s) between these slots, then the new station randomly and uniformly selects one of the empty slots and transmits in that slot at the next backoff period. If there is no empty slot, then the station doubles the $V(d)$, observes the remaining slots of the new $V(d)$ and use the same process explained above. After acquiring a slot, the station keeps transmitting in that slot. Upon receiving a beacon frame, the station initializes its *approaching process* as the other stations do and may have a lower backoff value, bypassing the empty slots. It should be noted that if the network is in *stable-state* and a new station enters the medium, due to the factor x_e , the new station does not cause collisions. It simply acquires one of the empty slots left for the entrance of new stations. With the entrance of new stations, if all the slots in a period become full, new $V(d)$ is

reassigned synchronously as all stations will be aware of the overloaded situation at the same $V(d)$ period. By reassigning the $V(d)$, BCCA allows new stations to enter the medium anytime without causing collisions. This procedure lets stations maintain their *collision-free state*.

In BCCA, $V(d)$ values can only be the multiples of 4, such as 4, 8, 12 and more. In case all the slots in a backoff period are full, since the value of x_e cannot be zero, stations synchronously reassign their $V(d)$ simply by adding 4 to it. In this case, x_e will be 4.

$$V(d)_{new} = x_t + x_e \quad (5.20)$$

where x_e is set to the value that makes the $V(d)_{new}$ multiples of 4, such as if $x_t = 3$, $x_e = 1$ or if $x_t = 5$, $x_e = 3$, and so on. Furthermore, if stations on the medium have a lower deterministic backoff period than a station entered the medium recently, the new station uses an *adaptation phase* in order to allow itself to have the same backoff period with the other stations on the medium. In this phase, stations observe each deterministic backoff period whether there is a repetition at the slot occupancies in a $V(d)$ (e.g. 5 stations and 3 empty slots may repeat their behavior twice in a $V(d)$ that is equal to 16). If other stations have a repetitive slot occupancy in three consecutive $V(d)$ s, the new station reassign its $V(d)$ by dividing it by two. Algorithm 3 in the Appendix shows the pseudo-code of BCCA that is implemented to stations in a distributed manner.

where $V(d)$ is the deterministic backoff period. a_p is an *adaptation-phase* counter for a station that has a higher deterministic backoff period than the rest of the stations. m_e is the maximum number that a station needs to converge to a smaller backoff period. x_t is the estimated number of stations on the medium and computed as the number of slots occupied in a $V(d)$. x_e is the number of new stations possible to enter the medium. *backoff* is the backoff counter and V_f is the virtual frame counter, respectively.

5.3. Collision-free Adaptation of Unsaturated Stations

As in UCFA, in order to prevent the collision probability of unsaturated stations and to maintain the collision-free state, in BCCA, unsaturated stations holds *virtual frames* if they do not have a frame in their queue after the last transmission. The only job of these virtual frames is to count their deterministic backoff values down whenever they sense the medium idle after the DIFS intervals as if they actually have a frame in their queue. When the backoff values of virtual frames become zero, if there is still no frame in their queue to transmit, clearly, it does not transmit any frame but it begins to recount the deterministic backoff value again until a new frame is inserted into the queue. Whenever stations have a new frame in their queue to transmit, then they use the same remaining backoff value of their virtual frame and count down from the place it left. By this way, unsaturated stations will be also ideal to use a deterministic backoff value $V(d)$.

5.4. Adaptation to the EDCA: Traffic Prioritization

IEEE 802.11 Enhanced Distributed Coordination Access (EDCA) offers differentiated service by providing different waiting times for different traffic priority levels. Three metrics have to be examined for the traffic differentiation, namely, the Arbitration Inter-Frame Space (*AIFS*), Transmit opportunity (*TXOP*) and Contention Windows (CW_{min} and CW_{max}). EDCA provides contention-free access to the channel for a period called a Transmit Opportunity (*TXOP*). When *TXOP* is used, stations are allowed to transmit multiple frames. Since *TXOP* does not have any impact in the contention procedure, application of *TXOP* is a valid option for traffic differentiation in BCCA. However, *AIFS* is a method of prioritizing one access class over the other by modifying the time of stations that have to listen to an idle channel before decrementing the backoff counter. Therefore, utilization of *AIFS* violates the assumption that all the stations decrement their backoff simultaneously. Therefore, it is impractical in BCCA. Ultimately, traffic differentiation can be also achieved by choosing different deterministic backoff values for high and low-priority levels, CW_{high} and CW_{low} , respectively. If values of the contention windows are selected as powers of two (CW_{high} is an integer divisor of CW_{low}), adaptation of

different deterministic values allows stations to maintain the collision-free state. Therefore, it is also a valid option for BCCA.

5.5. Coexistence of BCCA and DCF

Since the deterministic backoff value in BCCA is computed adaptively using the empty and busy slots in the network, during a period, medium will have enough slots for both BCCA and DCF stations. In this case, stations implemented by BCCA will virtually try to capture some of the slots and stations implemented by DCF will randomly select some of slots. Collisions will be mainly between BCCA and DCF stations since BCCA stations will try to capture the empty slots of the last period. Thus, collision probability of coexistence situation of BCCA and DCF stations will be lower than the probability of collision of standard DCF stations.

6. SMART CHANNEL SCANNING WITH MINIMIZED SERVICE DISRUPTIONS

In IEEE 802.11 WLANs, channel scanning is the dominant factor that mostly affects the handover delay and hence, it has to be reduced to provide seamless handover. In this context, we first examine the total expected scanning time for each scanning approaches, classifying them into the following groups: *passive*, *active*, *selective*, and *unicast*. In *passive scanning*, the total time needed to scan N_{ch} channels can be computed as follows,

$$T_{ps} = N_{ch}(T_{bwt} + T_{switch}) \quad (6.1)$$

where T_{bwt} is the beacon waiting time and T_{switch} is the channel switching delay. T_{switch} is Network Interface Card (NIC) dependent and may take up to a few milliseconds.

In *active scanning*, the total time needed to scan N_{ch} channels can be computed as follows,

$$T_{as} = \alpha(T_{p_req} + T_{MinChTime}) + \beta(T_{p_req} + T_{MinChTime}) + N_{ch} \times T_{switch} \quad (6.2)$$

where α and β is the number of non-active and active channels, respectively and the sum of them is equal to N_{ch} . T_{p_req} is the probe request time. $T_{MinChTime}$ and $T_{MaxChTime}$ are two major parameters in *active scanning* and represent the minimum and maximum channel waiting times, respectively. *Active scanning* is usually faster than *passive scanning* since stations probe candidate APs without waiting for the beacons.

If a station somehow has the information of currently active channels, then the station scans only those active channels. This procedure is known as *selective scanning* and the time needed to scan N_{live_ch} active channels can be computed as,

$$T_{ss} = N_{live_ch}(T_{p_req} + T_{MaxChTime} + T_{switch}) \quad (6.3)$$

Clearly, *selective scanning* reduces the total scanning time compared to the *active scanning*. However, the reliability is dependent on the information provided by the source (i.e., a central server).

Additionally, if a station somehow has the information of currently active channels and the SSIDs of APs in each active channel, the station sends unicast probe requests to each individual AP. This procedure is known as *unicast scanning* and the total time needed to scan N_{live_ch} active channels can be computed as,

$$T_{us} = N_{live_ch}(T_{p_req} + T_{p_res} + 2T_{SIFS} + 2T_{ACK} + T_{switch}) \quad (6.4)$$

where T_{p_res} is the probe response time. *Unicast scanning* is more reliable than *selective scanning* since the probe requests are unicasted and acknowledged by the AP. However, this approach may be inefficient if there are multiple APs in the same channel since station has to send individual probe requests to each AP.

In IEEE 802.11 WLANs, after a successful frame transmission, stations keep monitoring the channel until it is measured idle for a Distributed Inter-Frame Space (DIFS) interval. After the DIFS interval, backoff process starts. Whenever the backoff counter of a station reaches zero, the station transmits its frame in the next slot time. Delay-sensitive applications (e.g. VoIP) have unsaturated traffic condition (i.e., stations do not always have a frame in their queues). A station that has no frame in its queue waits until a frame is inserted into its queue and then, activates its backoff counter to access the channel. As an example, G.711 is a narrowband voice codec that provides toll-quality audio at 64 Kbit/s with 20 milliseconds packetization interval in default. This roughly means that a new frame will be inserted into the queue of the station once in every 20 ms.

In order to perform background channel scanning with minimized communication interruptions and to improve the throughput performance of stations, our proposed algorithm waits for the station to transmit its frame successfully before switching to another channel for the background scanning. After the successful frame delivery, the station switches to other channels and perform background scanning

during a pre-computed interval, T_{scan} . With the end of this interval, the station switches back to its channel and continue to frame transmission and reception. In the proposed scheme, T_{scan} is equal to the packetization interval for UDP stations. For TCP stations, T_{scan} is adaptively computed as the average transmission time of the last transmitted three frames. Additionally, the proposed scheme lets TCP stations to have a small backoff value ($CW = [0, 7]$) after the channel scanning phase. Thus, the proposed scheme limits the channel access time of TCP stations that have performed channel scanning recently. Since the computation is made adaptively by the average transmission time of the last transmitted three frames or by the packetization interval, channel scanning will mostly be performed in a time the station will not transmit a frame (especially for the UDP traffic). Therefore, this procedure will minimize the communication interruptions of stations and improve the overall throughput with smart distribution of *channel-scanning* and *frame-delivery* phases.

Additionally, periodic background scanning interval and triggering condition (RSSI threshold) are tunable vendor specific parameters and optimization of the scanning interval is difficult. The scanning overhead will increase if the scanning interval is short. However, if it is too long, a station may not detect the signal strength changes of neighboring APs on time and may miss the opportunity to re-associate with a better AP. In order to perform channel scanning at the right time and frequency, movements of stations are detected by an accelerometer in our smart scanning method. Accelerometer is widely used as a motion sensor in the latest smart devices. Global Positioning System (GPS) can also locate and trace the movement of stations. However, GPS cannot be used for indoor environments and consumes approximately ten times more energy than an accelerometer.

Flowchart of the proposed scheme is shown in Figure 6.1. The methodology of the proposed scheme can be classified into three phases: *activation*, *monitoring* and *updating phase*. *Activation phase* is initialized whenever a station powers its WLAN interface on. At first, the station performs *passive scanning* for all possible channels to store the beacon arrival times of each AP and to make use of them in the upcoming background scanning phases. During the initial *passive scanning*, the station stores neighboring APs in a hit-channel list (HCL). The list holds information detected during the scanning process for each AP, such as the basic service set identifications (BSSID), received signal strength indicators (RSSI), beacon intervals (BI), and beacon arrival times (BAT). After the *passive scanning* is completed, the

station associates with the AP that has the highest RSSI.

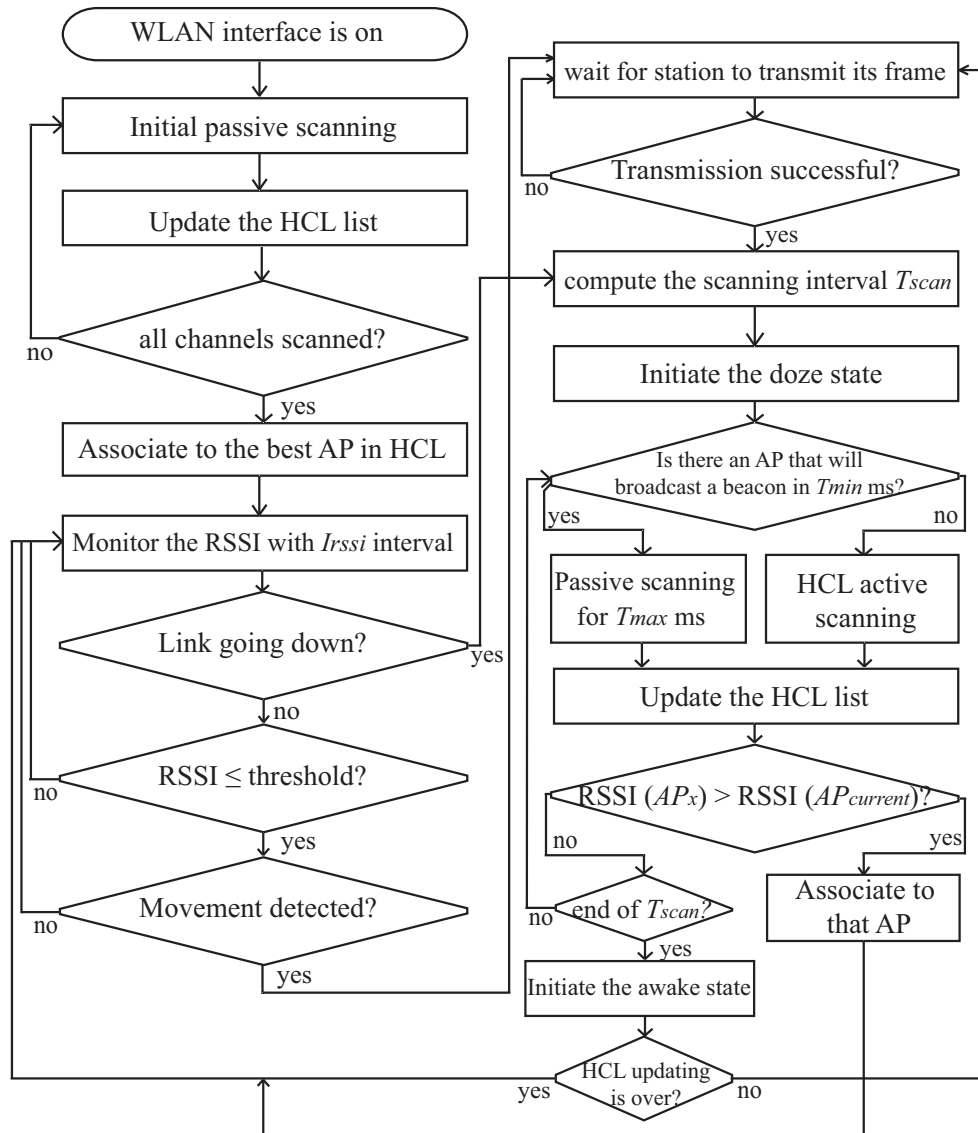


Figure 6.1: Flowchart of the proposed smart channel scanning scheme.

In the *monitoring phase*, the station periodically monitors the RSSI of the associated AP. If the RSSI is lower than a pre-defined threshold and movement is detected, then the station initializes its *updating phase*. In this phase, if the station has already a frame in its queue, then the station waits for its frame to be transmitted successfully. If the transmission is successful, the station computes the scanning interval, T_{scan} as explained in the beginning of this section. Before switching from the current channel to target channels, the station initiates its *doze state*. This allows the AP that is associated with the station to buffer the frames destined to the station. The

station is now ready for the background scanning and performs the following procedure.

The station first checks whether there is any channel expected to transmit a beacon frame in T_{min} interval (i.e., assigned as 5ms in the simulations) looking at the BATs in the HCL. If there is an expected beacon frame in T_{min} interval, the station switches to that channel and waits for the beacon frame for maximum T_{max} interval (i.e., assigned as 10ms in the simulations). If there is no channel expected to transmit a beacon in T_{min} interval, the station performs *active scanning* in descending order of RSSI values of target APs in the HCL. In the *active scanning*, if a probe-response is not received, the station waits for $T_{MinChTime}$. If a probe-response is received, the station waits for $T_{MaxChTime}$ unless there is a channel expected to transmit a beacon frame in T_{min} interval. As long as the channels are scanned, the station updates its HCL. Depending on the magnitude of the T_{scan} , several channels can be scanned within a single smart scanning interval. The channels in the order are continuously scanned until T_{scan} period is completed. Alternation of *channel-scanning* and *frame-delivery* phases is performed until all of the candidate channels are scanned. Therefore, the proposed smart channel scanning procedure may be composed of several smart scanning intervals.

During the *scanning-phase*, if an AP that has a higher RSSI than the associated AP is detected, the handover process begins immediately. Additionally, if there is no AP available in the HCL when the background scanning process is triggered, the station scans all the channels using the legacy method specified in the IEEE 802.11 standard.

After the channel scanning, the station switches back to its channel, transmits a PS-Poll frame to its AP to get the frames buffered during the last T_{scan} interval, and changes its state from *doze state* to *awake state*. With the proposed method, connection disruption time and the time spent for the periodic background channel scanning can be reduced dramatically by suppressing those unnecessary scanning attempts.

As an example, Figure 6.2 shows the proposed smart channel scanning procedure of a VoIP station. As shown in the figure, *channel-scanning* phases are initiated after frame deliveries and performed for 20ms intervals. The station first switches to the *channel-D* as the *channel-D* is supposed to transmit a beacon frame in T_{min} interval. After receiving a beacon from the *channel-D*, the station performs an

active scanning on the *channel-C* which has the highest RSSI in the HCL. *Active scanning* in *channel-C* is not performed for $T_{MaxChTime}$ since *channel-A* is expected to transmit a beacon frame in T_{min} interval. Therefore, after receiving the probe response frame from the *channel-C*, the station switches to the *channel-A* and receives the beacon frame of the *channel-A*. Next, the station switches to the *channel-F* since it is the channel with the highest RSSI among the un-updated channels in HCL. The station transmits a probe request and waits for the probe response until the end of the T_{scan} period. However, the station does not receive any probe response and then switches back to its serving channel and performs its *frame-delivery* phase. After the successful frame transmission, the station switches to the *channels G, B, F and E*, respectively, receives the beacon frames and updates the HCL.

It should be noted that in the *scanning phase 3*, the station waits for $T_{MaxChTime}$ since there is no channel expected to transmit a beacon frame in T_{min} interval during the T_{scan} period. Therefore, the station observes the *channel-E* whether there is another AP on the same channel that can be inserted in the HCL. Consequently, the station scans all the channels in three smart scanning phases, updates its HCL and switches back to its serving AP. In this context, the total scanning time is the sum of the number of T_{scan} periods and the time spent for the *frame-delivery* phases.

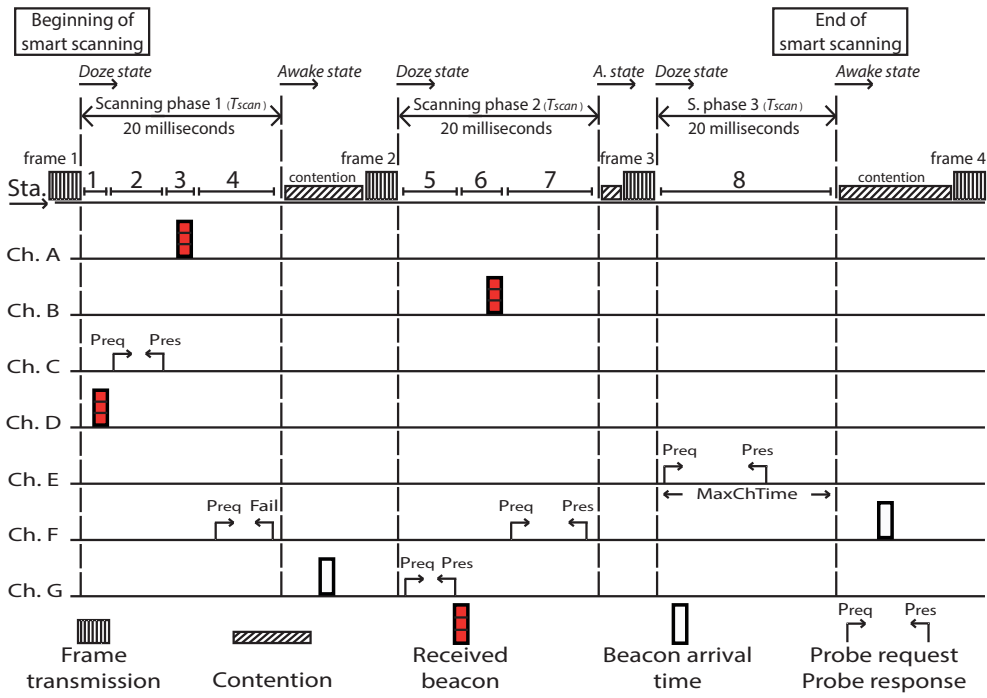


Figure 6.2: Smart channel scanning procedure of a VoIP station.

7. EVALUATION OF THE PROPOSED SCHEMES

7.1. Evaluation of Collision-free Channel Access - UCFA

Performance evaluation of UCFA is analyzed by extensive simulations using the OMNET simulator and a custom-made object-oriented event-driven simulator software written in C++. Some of the parameters used to calculate formulas and implement the IEEE 802.11 MAC layer are shown in Table 7.1 and Table 7.2. We implemented our own TCP/VoIP/Video traffic generator and evaluation tool that creates realistic data and multimedia packet streams using the parameters in Table 7.1 and Table 7.2. During the simulations, TCP frames are implemented as saturated frames. Voice and video frames are implemented as unsaturated frames and queued in 20ms intervals. In order to reach saturation conditions, offered load has been set greater than the maximum achievable throughput.

Table 7.1: Timing units and values for IEEE 802.11b.

Name	Value	Unit
Slot time	20	us
Message size	20 - 1000	Byte
Bit rate	1 - 11	mbps

Table 7.2: IEEE 802.11b parameter values used at simulations.

Name	Length (bits)	Bit rate (Mbps)	Time (us)
DIFS	50	1	50
Preamble	192	1	192
Header	240	11	21,81
Data (TCP)	8000	11	727,27
Voice (UDP)	1280	11	116,36
Video (UDP)	3200	11	290,9
CRC	32	11	2,909
SIFS	10	1	10
ACK	112	2	56

Simulation environment is shown in Figure 7.1. It is a hotspot scenario that is composed by one 802.11 BSS with wireless stations and one AP connected to wired network. A successful transmission is confirmed by the corresponding

acknowledgement from the receiver. An unsuccessful transmission is caused either by a collision or a frame error.

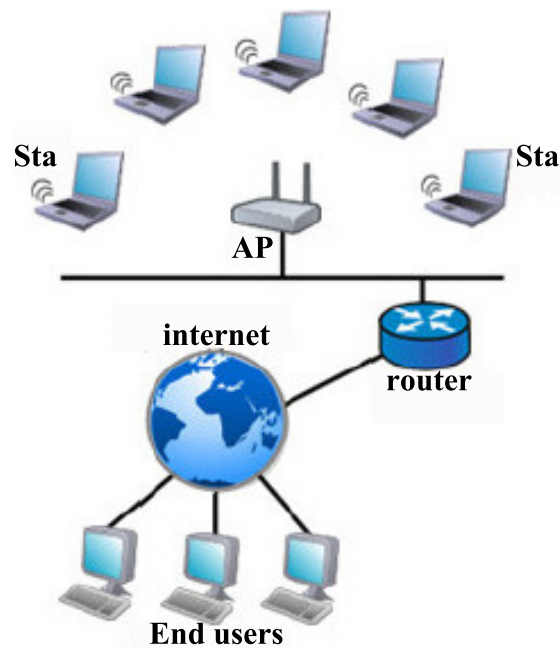
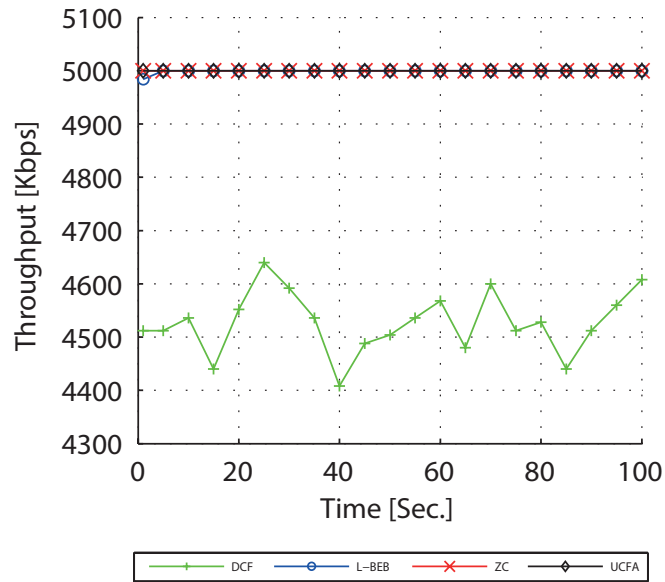


Figure 7.1: IEEE 802.11 network communication scenario.

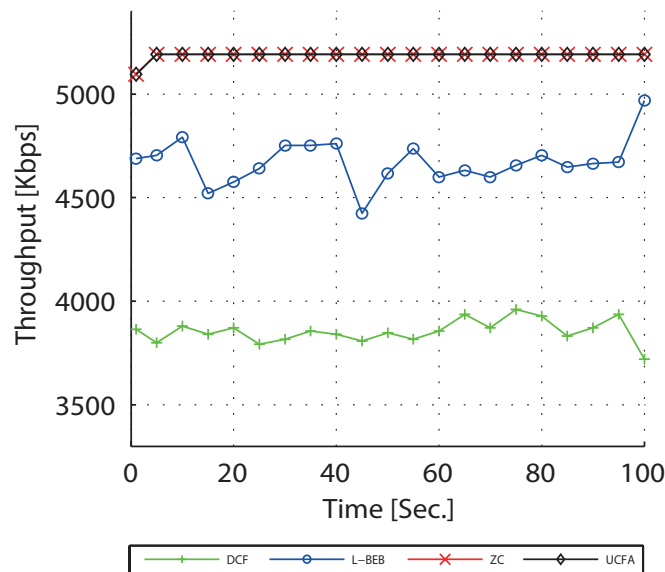
Six different cases are implemented and examined in this section. Our proposed scheme UCFA is compared to the standard DCF, EDCA, L-BEB and ZC throughout all the scenarios. The first case is based on ideal channel conditions. We examine only the presence of various number of data stations in the network. In the second case, we examine the combination of saturated and unsaturated stations. Third, we examine the effect of new stations that enter the medium and the performance fluctuation when the number of stations exceeds the deterministic backoff value. Fourth, we examine the effect of hidden nodes on the throughput. Fifth, coexistence performance of the UCFA and other hybrid approaches with the standard DCF for both saturated and unsaturated traffic is examined. Finally, fairness, channel efficiency and QoS support for various traffic types are analyzed in the sixth case.

In the first case, two different scenarios are implemented and examined. At first, system performance of high deterministic backoff value selection for limited number of data (saturated) stations is analyzed under ideal channel conditions. Second, system performance when the deterministic backoff value is equal to the

number of data stations is analyzed under ideal channel conditions. Figure 7.2 shows the throughput performance of UCFA against hybrid schemes and the standard CSMA/CA.



a)



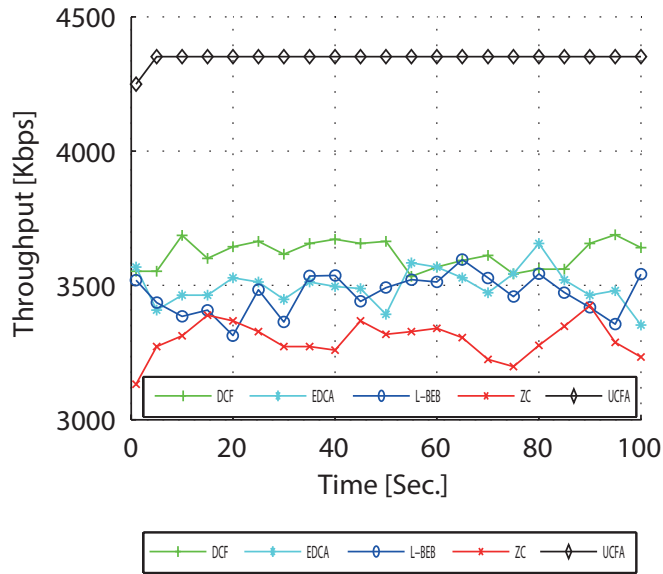
b)

Figure 7.2: Throughput values of related schemes when the channel is ideal and all stations are saturated. a) Number of stations is 4 and $V(d) = 16$, b) Number of stations is 16 and $V(d) = 16$.

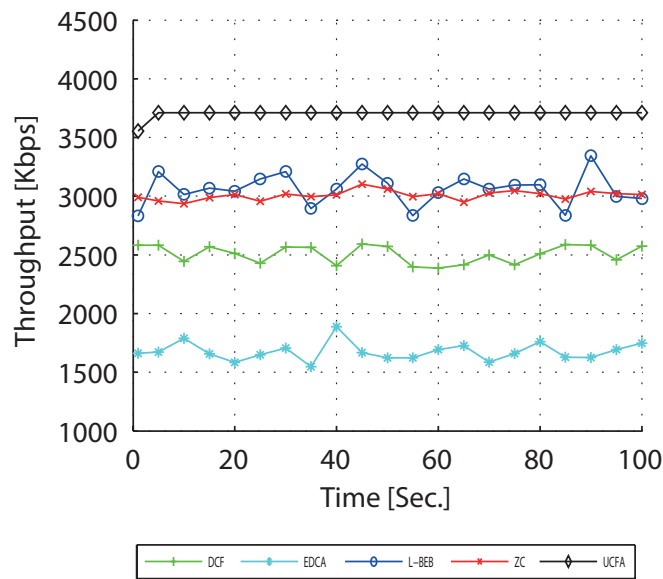
As shown in Figure 7.2-a, overall throughputs of UCFA and other hybrid schemes significantly outperform the standard DCF scheme. After the first second, throughput values of UCFA and other hybrid schemes during the simulation are the same. It is because, in saturated regime, hybrid schemes select a deterministic backoff after successful transmissions in a similar fashion. Thus, after all participating stations successfully transmit, stations acquire a particular slot and keep transmitting in that order. Therefore, the system enters its collision-free state. Besides, since there is no probability of a frame error for the ideal channel condition, UCFA and other hybrid schemes have the same amount of throughput after the collision-free state for this scenario. However, in CSMA/CA, collisions may occur anytime due to its random binary exponential backoff procedure and this leads to a significant decrease at the throughput.

As it is depicted in Figure 7.2-b, both UCFA and ZC have higher throughput than they have in Figure 7.2-a. It is because the number of stations is higher and it causes a more efficient usage of the medium since the proportion of the empty slots will be lower. However, on the contrary, the throughput of L-BEB decreases significantly compared to the Figure 7.2-a. It is because L-BEB selects a random backoff value after a collision as it is in CSMA/CA. Since the number of stations is high and equal to $V(d)$, the probability of converging to collision-free state for L-BEB is very low in this scenario. Additionally, since the collision probability increases in parallel with the number of stations on the medium, DCF has lower throughput as well.

In the second case, we examine the channel performance of heterogeneous network, which is composed of saturated and unsaturated stations, under the ideal channel condition. Throughput performance of UCFA against hybrid schemes and the standard CSMA/CA under various numbers of stations and deterministic backoff values is shown in Figure 7.3.



a)



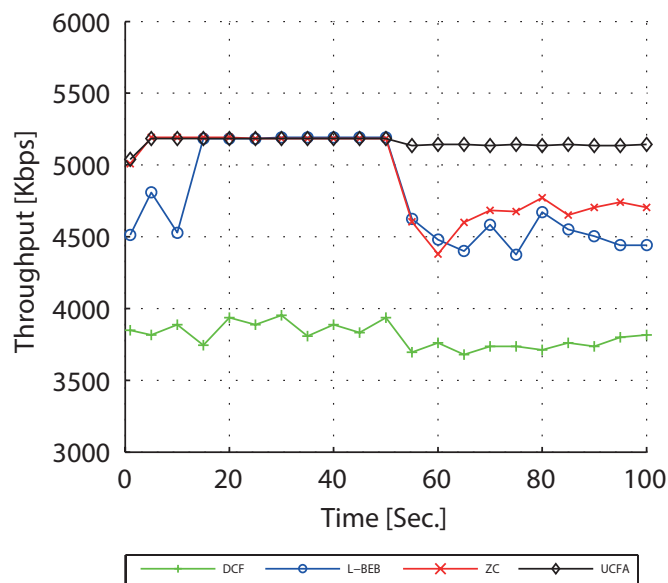
b)

Figure 7.3: Throughput values of related schemes when the traffic is a mix of voice, video and data stations. a) $V(d) = 8$, Nr. of voice, data and video stations are 3, 2, 3., b) $V(d) = 16$, Nr. of voice, data and video stations are 6, 4, 6.

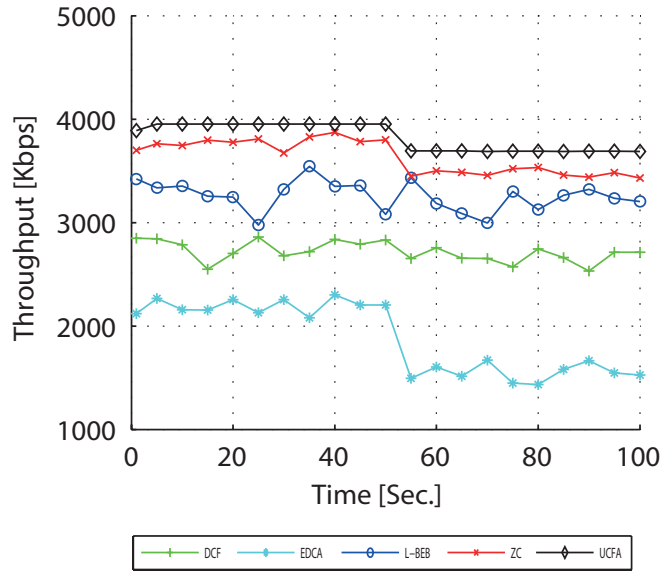
Since the virtual frame procedure of UCFA allows each station to transmit its frames in its own particular slot even in the presence of unsaturated stations, stations maintain their collision-free states and maximize their throughput in UCFA as it is shown in Figure 7.3. However, in other hybrid schemes, unsaturated stations may

collide if they have a new frame in their queue to transmit and select the same slot that seems empty at the same time. As a result of new collisions, the collision-free state is broken and it leads to performance degradation. In this case, L-BEB mainly has higher throughput than ZC. It is because, while L-BEB exponentially increases its backoff after a collision in order to decrease the proportion of collisions, ZC randomly selects the slots, which seem empty every time. Finally, due to assigning different priorities to different traffic types, EDCA provides higher QoS but lower throughput than the DCF.

In the third case, we examine the effect of new stations that enter the medium and the performance fluctuation when the number of stations exceeds the deterministic backoff value. Throughput performances of the standard CSMA/CA, UCFA and the other hybrid schemes for both saturated stations and the combination of saturated and unsaturated stations are shown in Figure 7.4.



a)



b)

Figure 7.4: Throughput values of related schemes. 3 new stations enter the medium after the 50th second. a) $V(d) = 16$ and the Nr. of data stations is 15, b) $V(d) = 16$ and the Nr. of voice, data, video stations are 5, 5, 5.

As it is seen in Figure 7.4-a, UCFA and other hybrid schemes quickly converge to the collision-free state and then have a very efficient usage of the medium during the first 50 seconds since there is no collision and only one out of 16 slots is empty (number of stations is 15). However, afterwards, three new stations enter the medium and it leads the number of stations to exceed the deterministic backoff value. Other hybrid schemes assign random backoff values for these new stations and cause new collisions during the excessive usage of the medium. In UCFA, these new three stations first wait for the new beacon frame before attempting to transmit, calculate the new $V(d)$ as the existing stations do, then observe the previous $V(d)$ periods and select one of the empty slots to start transmitting. Since each station calculates the $V(d)$ periodically upon receiving each beacon frame, stations notice the excessive usage (all the slots in a $V(d)$ are busy) of the network after the 50th second and double the $V(d)$ in UCFA. Therefore, the new $V(d)$ becomes 32 and stations converge to a new collision-free state afterwards again.

Due to the presence of unsaturated stations, hybrid schemes never converge to the collision-free state as shown in Figure 7.4-b. However, in UCFA, collision-free state is maintained by the virtual frame procedure of unsaturated stations.

In the fourth case, we examine the effect of hidden nodes on the throughput. Figure 7.5 shows the average throughput values of related schemes (DCF, RTS/CTS and UCFA) with respect to the number of hidden nodes on the medium. We set the total number of nodes (including the hidden nodes) on the medium as 16 and 32 and plotted the average throughput values (saturated traffic) one by one for each scheme by varying the number of hidden nodes. Each sample in the figure is the average of 10 simulation runs, each of which has an interval of 100 seconds.

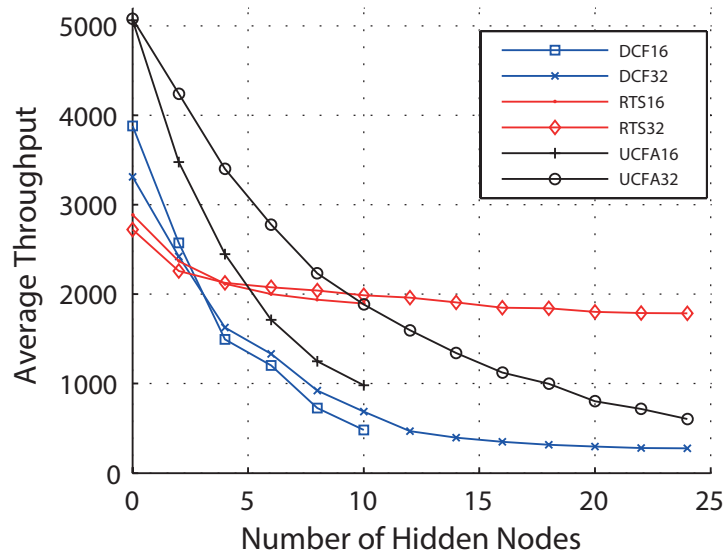
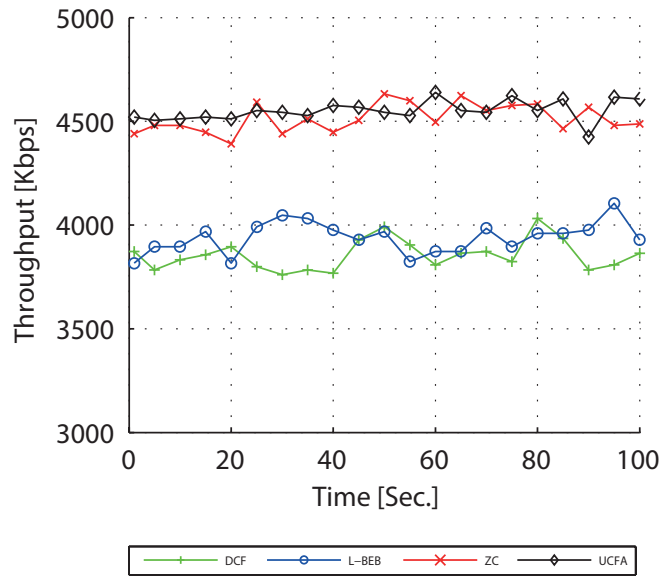


Figure 7.5: Average throughput vs. Number of Hidden nodes for related schemes.

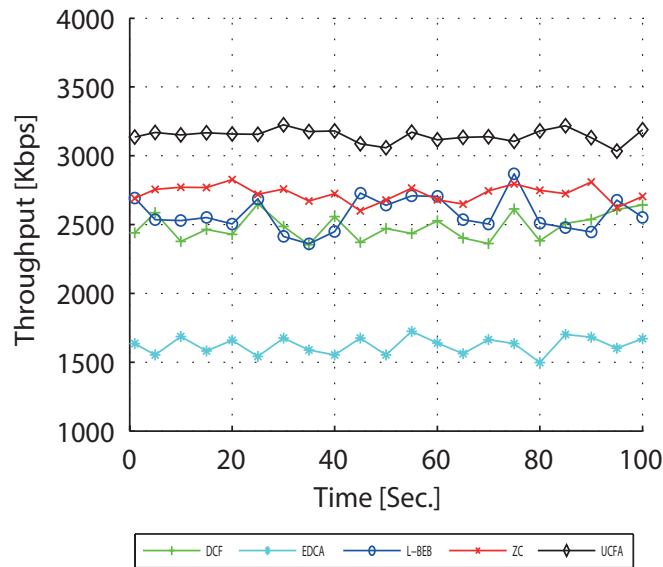
As is shown in Figure 7.5, number of hidden nodes on the medium is directly related to the achievable throughput. An increase on the number of hidden nodes decreases the average throughput dramatically for both UCFA and the standard DCF. RTS/CTS frames can minimize the negative effects of hidden nodes. However, if there is no hidden node or there are only few of them, UCFA and the standard DCF outperform the RTS/CTS solution.

In the fifth case, coexistence performance of the UCFA and other hybrid schemes with the DCF is examined. In this case, each medium is implemented as half of the stations are using the UCFA or one of the hybrid schemes and the other half are using the standard DCF. Overall throughput performance of the coexistence

situation for both saturated and unsaturated traffic under non-ideal channel condition is shown in Figure 7.6. Frame error rate is considered as 1% in this case.



a)

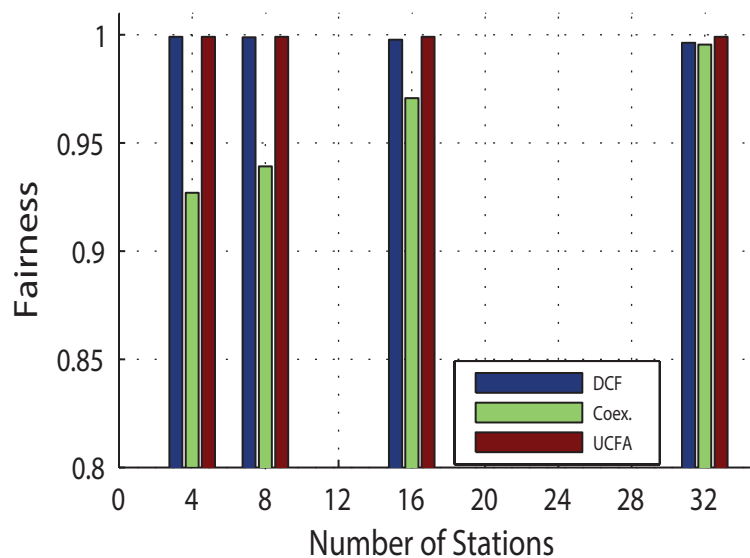


b)

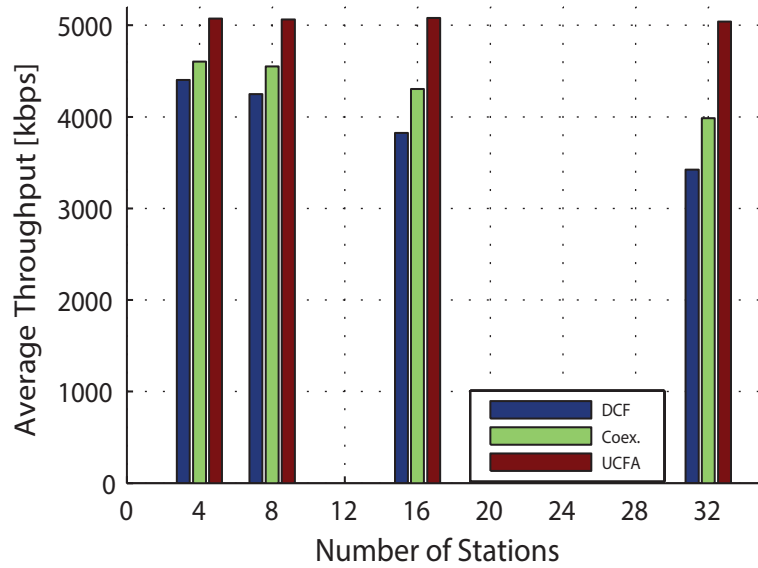
Figure 7.6: Coexistence throughputs of UCFA-DCF, ZC-DCF, L-BEB-DCF and DCF itself when channel error-rate is 1%. a) $V(d) = 16$ and the number of data stations is 16, b) $V(d) = 16$ and the # of voice, data, video stations are 6, 4, 6.

Coexistence with the DCF may cause collisions anytime in network due to the random binary exponential backoff procedure of the DCF. This randomness prevents UCFA or other hybrid schemes to converge to the collision-free state and hence, decreases the channel efficiency. However, as shown in Figure 7.6, UCFA-DCF coexistence situation mainly has higher throughput than other coexistence approaches during the simulation due to the real-time adaptation of UCFA as explained in the previous sections.

Figure 7.7-a shows the simulation results of the fairness index of the DCF, UCFA and their coexistent scenario for saturated stations (all stations have same weight in the simulations). In the figure, we observe that both UCFA and the standard DCF schemes are highly fair. UCFA provides the fairest network allocation independent of the number of stations on the medium. We also observe that the coexistent scenario fails at fair allocation of the network especially when the number of stations on the medium is low. However, coexistent scenario increases its fairness in parallel with the increase in the number of stations. When the number of station is equal to or higher than 32, coexistent scenario provides fairer network allocation than DCF stations. It should also be noted that the coexistent scenario may not be as fair as the DCF or UCFA schemes, but it provides higher throughput than the standard DCF scheme as shown in Figure 7.7-b.



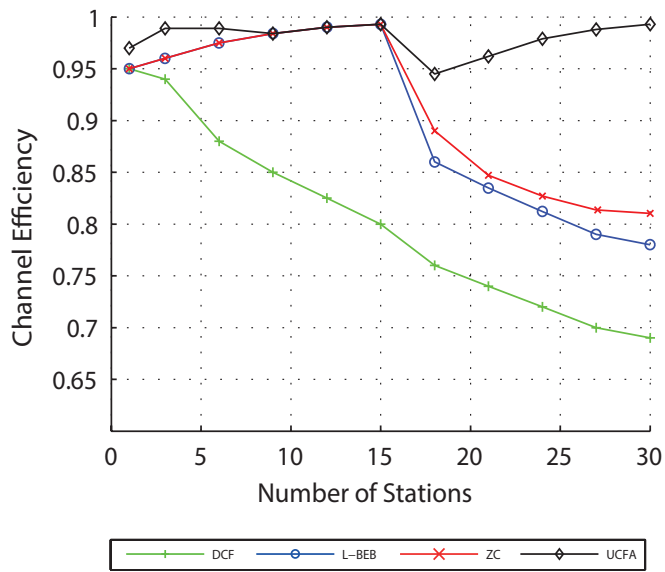
a)



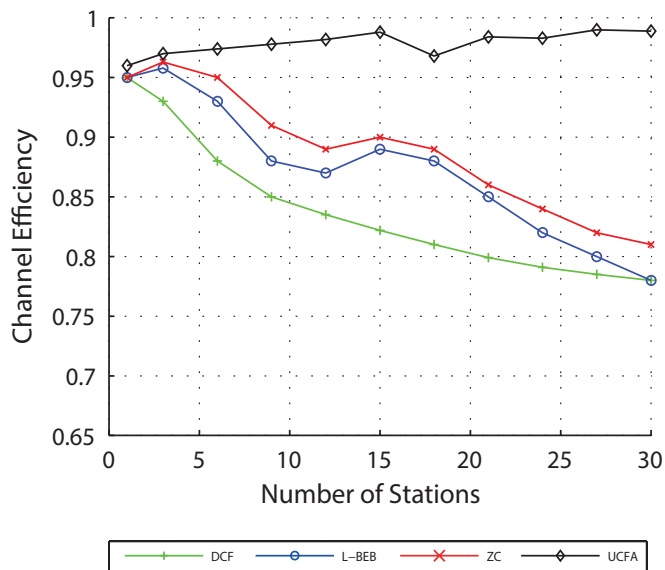
b)

Figure 7.7: Fairness & average throughputs for DCF, UCFA and Coexistent stations. a) Fairness vs. Number of stations, b) Average throughput values vs. Number of stations.

Figure 7.8 shows the channel efficiency of the medium when UCFA and related schemes are implemented for saturated stations and the combination of saturated and unsaturated stations under non-ideal channel conditions. As depicted in Figure 7.8-a, when the medium consists of only data stations, both UCFA and other hybrid schemes converge to the collision-free state. However, when the number of contending stations exceeds the deterministic backoff value $V(d)$, collisions occur and this leads to the performance degradation. In this case, only the UCFA scheme re-converges to its collision-free state by rearranging (doubling) the $V(d)$.



a)



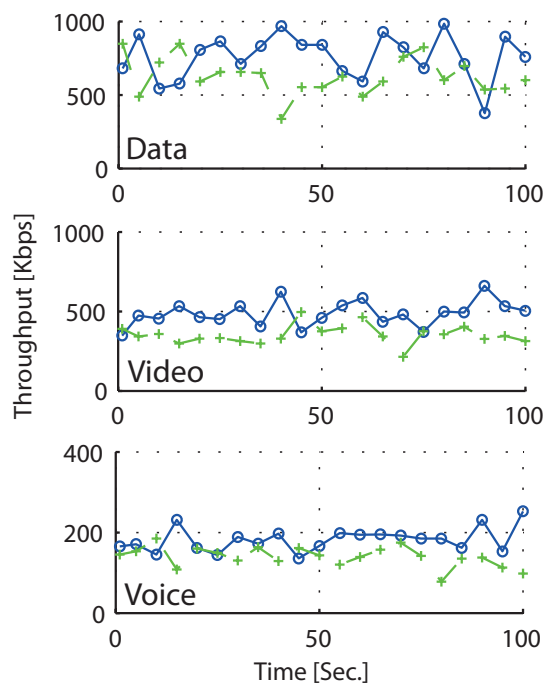
b)

Figure 7.8: Channel efficiency vs. Number of stations when $V(d) = 16$. a) Channel efficiency vs. Number of data stations, b) Channel efficiency vs. Number of voice, data, and video stations.

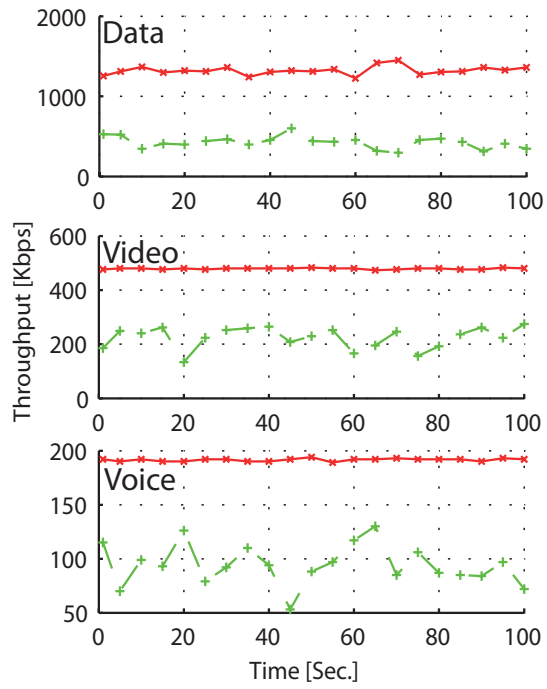
As it is seen in Figure 7.8-b, when the medium has both saturated and unsaturated stations, the standard DCF and other schemes have collisions. However, UCFA converges to collision-free state and maximize its channel efficiency by making use of virtual frame initialization. Doubling the $V(d)$ causes a decrease on the channel efficiency of UCFA at first since the number of empty slots increase.

However, in time, channel efficiency increases in parallel with the number of stations on the medium.

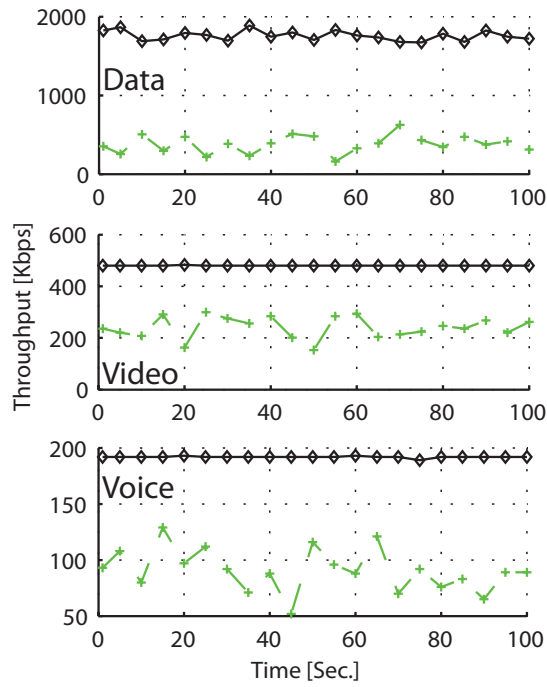
Figure 7.9 depicts the coexistence throughput values of voice, video and data frames separately for the L-BEB, ZC and UCFA. Since the UCFA-DCF coexistence situation mainly has higher throughput than other approaches, multimedia stations implemented by UCFA have not only fewer collisions but also less delay and jitter. However, in ZC and L-BEB, more collisions occur and it causes higher fluctuations on the transmitting time of multimedia frames. Thus, multimedia frames suffer more delay and jitter.



a)



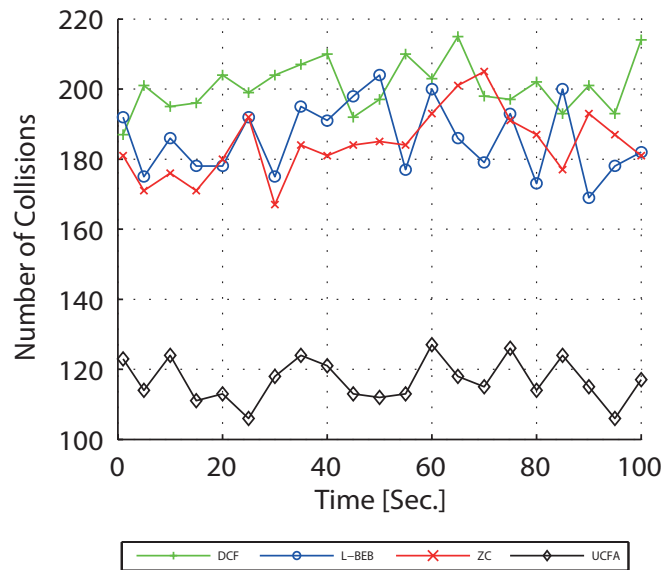
b)



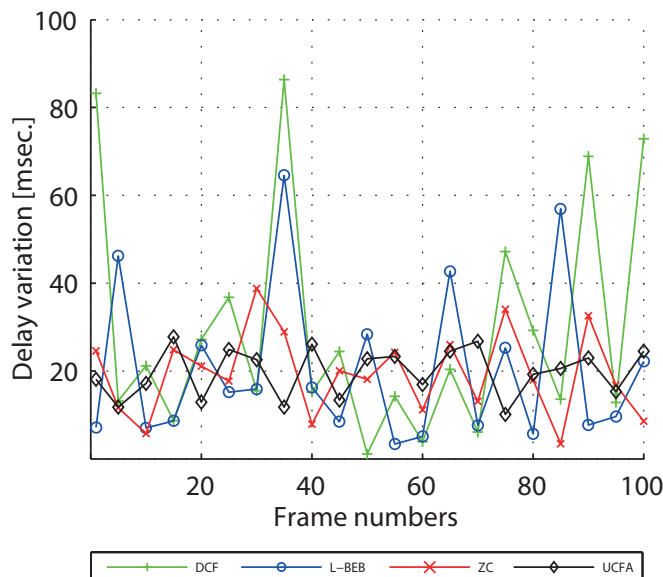
c)

Figure 7.9: Coexistence throughputs of related schemes when $V(d) = 16$ and Nr. of voice, data, video stations are 6, 4, 6. a) L-BEB – DCF, b) ZC – DCF, c) UCFA – DCF.

Figure 7.10-a shows the number of collisions in five-second intervals for UCFA-DCF, L-BEB-DCF, ZC-DCF and DCF coexistence situations. Since the network is a combination of saturated and unsaturated stations, collision probability of hybrid schemes is almost as high as the DCF itself. However, virtual frame procedure of UCFA prevents further collisions and lets collision probability be as low as possible even in coexistence situation.



a)



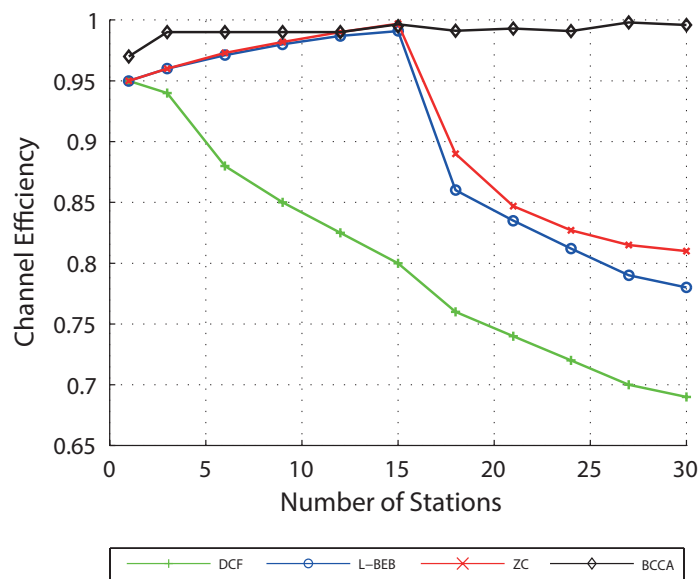
b)

Figure 7.10: Collision and Delay variation values for the Coexistent scenario. a) Number of collisions vs. Time, b) Delay variation of the first 100 transmitted frames.

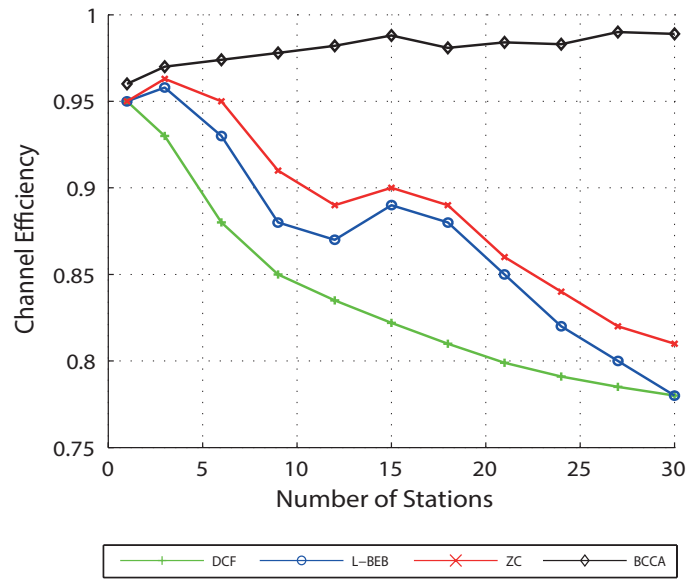
Figure 7.10-b shows the delay variation (jitter) of the first 100 transmitted frames. Since the packetization delay of G.711 voice codec is assigned as 20ms in the simulations, a delay variation at the receiver side near 20ms is desired. While UCFA always maintains the delay variation in between 11–29ms and support desired QoS, other hybrid schemes and DCF have higher delay variations that make the voice stations vulnerable since it may cause packet drops.

7.2. Evaluation of Beacon-based Channel Access - BCCA

Throughout the simulations, BCCA is compared with the standard DCF, L-BEB and ZC algorithms. In the first case, we examine only the presence of various number of data stations in the network. Figure 7.11 shows the channel efficiency of the medium when BCCA and related schemes are implemented with only saturated stations (Figure 7.11-a) and a combination of saturated and unsaturated stations (Figure 7.11-b), respectively under ideal channel condition.



a)



b)

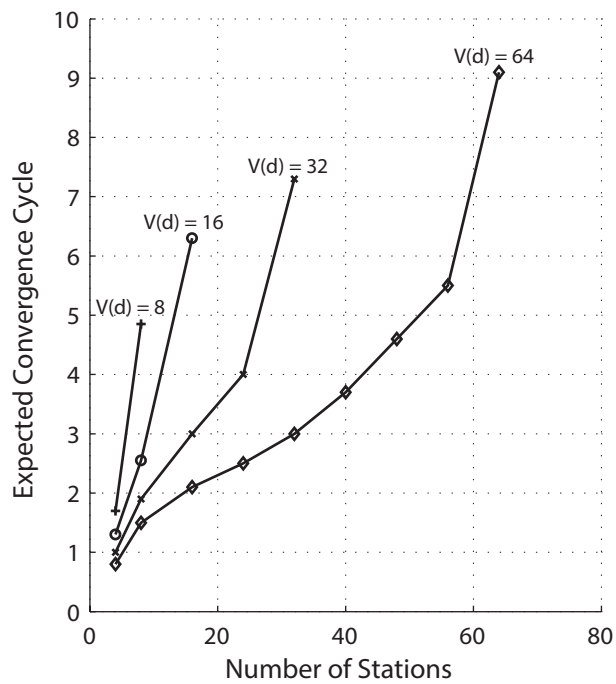
Figure 7.11: Channel efficiency vs. Number of stations. BCCA has an adaptive cycle length while L-BEB and ZC have a fixed cycle length; $V(d) = 16$. a) Saturated-state (data stations), b) Unsaturated-state (voice, data, video stations).

When the medium consists of data stations, both BCCA and other hybrid schemes converge to the *collision-free state* and have a high channel efficiency. However, since BCCA is the only algorithm that has the adaptive deterministic backoff selection according to the number of slots occupied by stations in each $V(d)$ period, BCCA has higher channel efficiency by adapting its $V(d)$ and hence reducing the idle slots. Furthermore, in related hybrid approaches, when the number of contending stations exceeds the pre-defined deterministic backoff value, collisions occur and it leads to performance degradation. In BCCA, due to the x_e metric, i.e., the number of new stations possible to enter the medium, stations rearrange the $V(d)$ period whenever a period is sensed full and prevent further collisions.

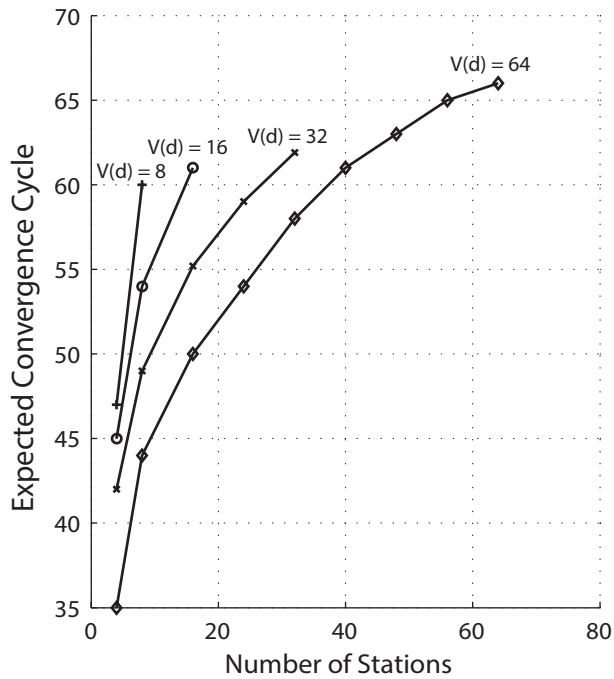
As shown in Figure 7.11-b, when the medium consists of both saturated and unsaturated stations, BCCA has slightly less channel efficiency than the condition when the medium consists of only saturated stations. It is because, the number of idle slots increases due to unsaturated stations do not always have a frame to transmit, and this leads to a slight decrease at the channel efficiency. However, since BCCA has a *virtual frame adaptation* to prevent collisions of unsaturated stations and maintain the *collision-free state*, channel efficiency remains high in comparison with

other hybrid schemes. Additionally, in BCCA, when the number of station increases, channel efficiency also increase due to the number of idle slots decreases.

Figure 7.12 shows the upper bounds of expected cycles for *collision-free state* and *stable state*, respectively. As it is seen in Figure 7.12-a, expected convergence cycles increase in parallel with the number of stations. Converging to the *collision-free state* is a fast process since stations randomly select one of the empty slots in the last $V(d)$ period and transmit in that slot. Moreover, the more idle slots means the quicker convergence. On the other hand, converging to the *stable-state* takes longer than converging to the *collision-free state* since stations wait for the beacon frames and then approach to their neighbor stations step by step, bypassing empty slots.



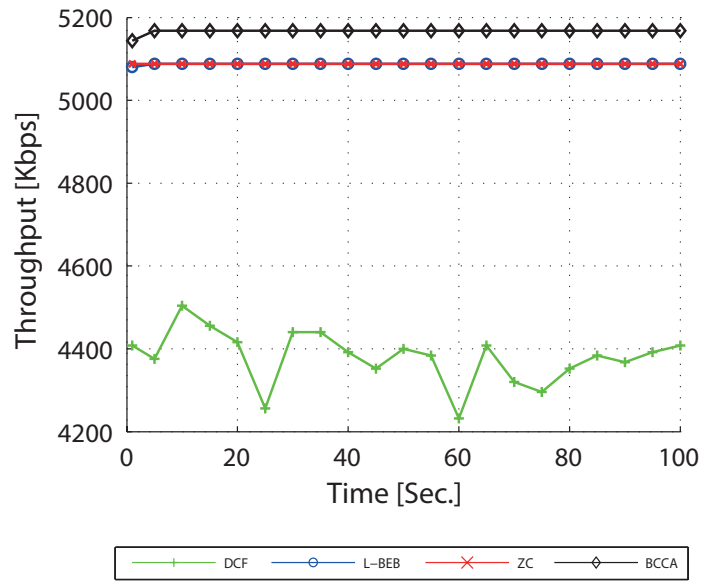
a)



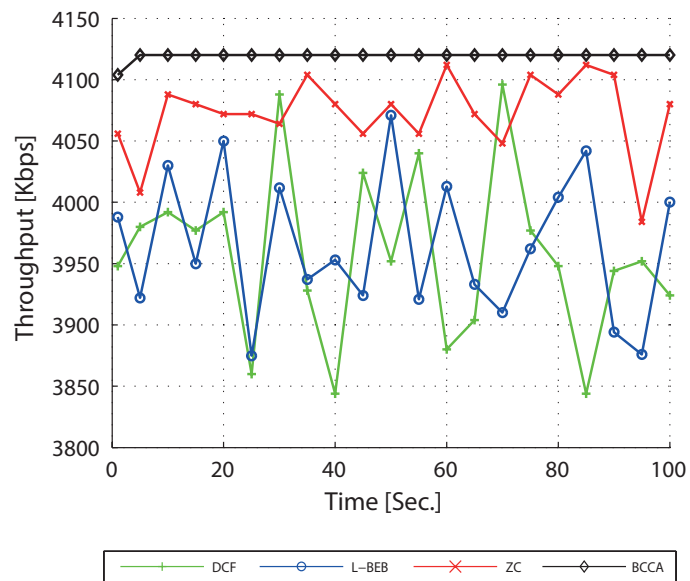
b)

Figure 7.12: Number of expected cycles for Collision-free state and Stable-state. a) Collision-free state, b) Stable-state.

Overall throughput for limited number of data and multimedia stations can be seen in Figure 7.13. When the medium only consists of data stations as shown in Figure 7.13-a, BCCA and other hybrid schemes significantly outperform the standard DCF scheme. In saturated regime, BCCA and hybrid schemes select a deterministic backoff value after successful transmissions in a similar fashion. Thus, after all stations successfully transmit, stations acquire a particular slot, keep transmitting in that order and system enters its *collision-free state*. Besides, since there is no probability of a frame error for ideal channel condition, hybrid schemes have the same throughput after the *collision-free state* for this scenario. However, BCCA has higher throughput since stations adapt their $V(d)$ periods in order to reduce the number of idle slots according to the number of stations on the medium (other hybrid schemes have only one deterministic backoff value that is 16). In CSMA/CA, collisions may occur anytime in network due to its random backoff selection and it leads to a significant decrease at the throughput.



a)



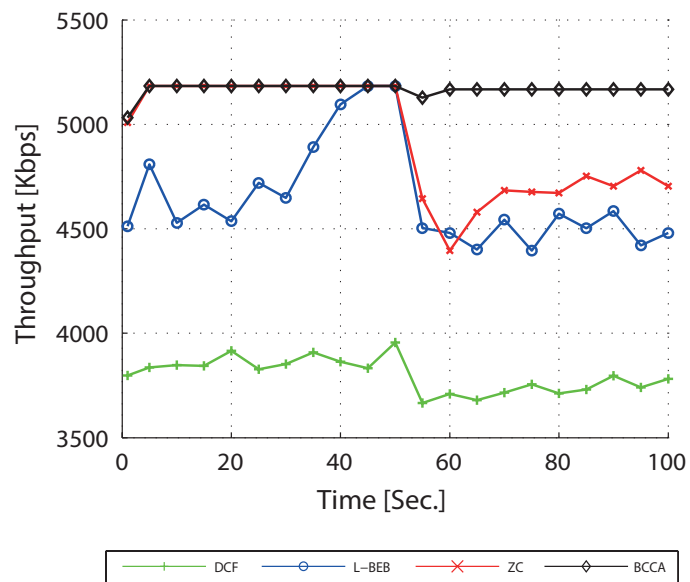
b)

Figure 7.13: Throughput values of related schemes. a) 6 saturated (data) stations, b) 2 voice, 2 video and 2 data stations.

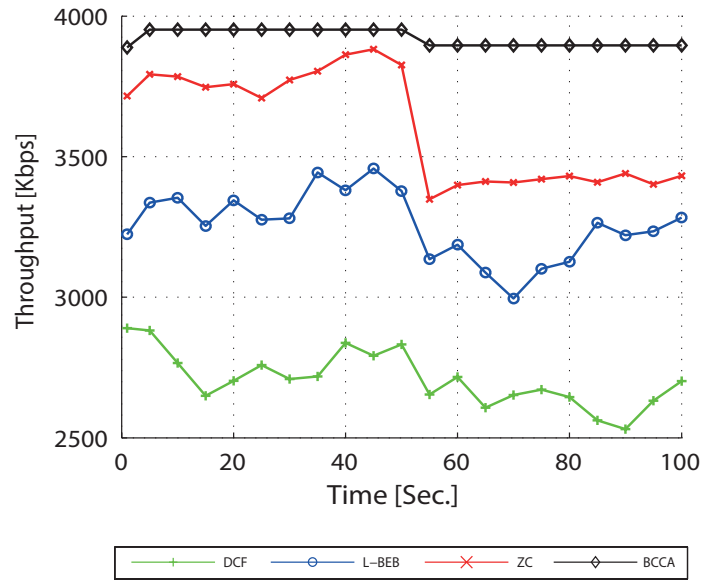
When the medium is a combination of saturated and unsaturated stations as in Figure 7.13-b, BCCA is the only scheme that maintains its *collision-free state* by initializing its *virtual frame adaptation*. Therefore, BCCA achieves high throughput by avoiding collisions while DCF and other hybrid schemes have less throughput,

being exposed to collisions, due to random transmission attempts of unsaturated stations.

Figure 7.14 shows the throughput values of related schemes when the medium is highly loaded with saturated stations (Figure 7.14-a) and combination of saturated and unsaturated stations (Figure 7.14-b), respectively. In this case, the number of stations on the medium is equal to 15, and three new stations enter the medium after the 50th second. Therefore, we examine the effect of new stations that enter the medium and performance fluctuation when the number of stations exceeds the deterministic backoff value.



a)



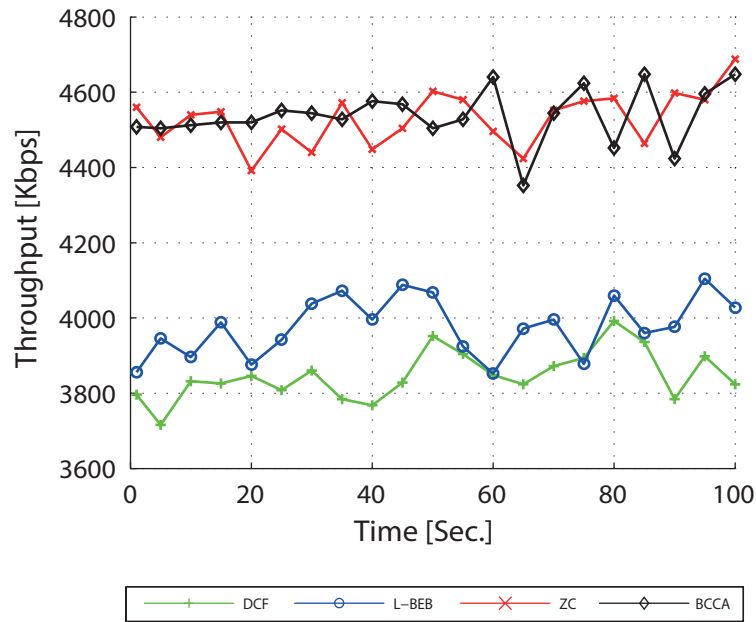
b)

Figure 7.14: Throughput values of related schemes. a) 15 saturated (data) stations, b) 5 voice, 5 video and 5 data stations.

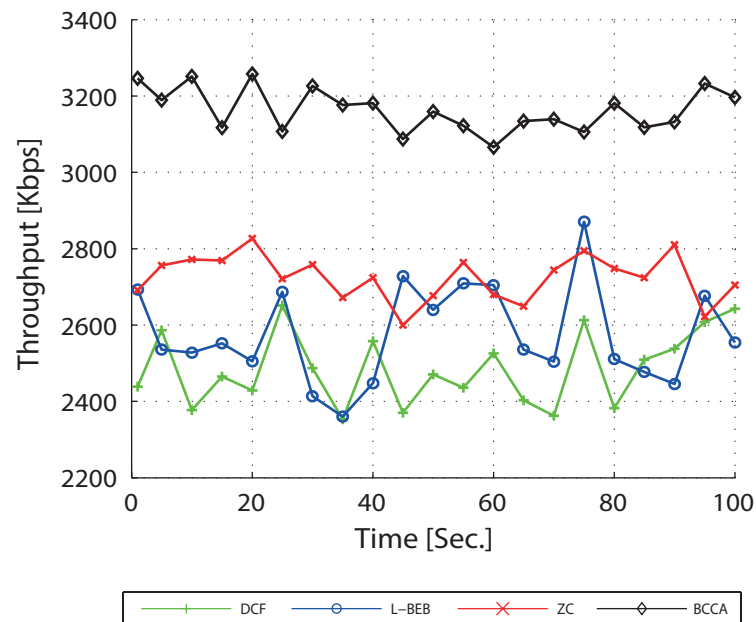
As it is seen in Figure 7.14-a, BCCA and ZC quickly converge to the *collision-free state* and then have a very efficient usage of the medium during the first 50 seconds since there is no collision and only one out of 16 slots is empty (number of stations is 15). However, in this case, converging to the *collision-free state* for L-BEB is not as fast as BCCA or ZC. It is because, in L-BEB, whenever a collision occurs, collided stations move back to their random behavior as the IEEE 802.11 DCF. Additionally, since the number of stations on the medium is 15 and the value of pre-defined deterministic backoff period is 16, the throughput of saturated stations after the *collision-free state* is the same for both BCCA, ZC and L-BEB. After the 50th second, L-BEB and ZC have collisions due to the number of stations on the medium (18) exceeds the pre-defined deterministic backoff value (16). Therefore, throughput values of L-BEB and ZC decreases after the 50th second. However, BCCA senses when all slots in a $V(d)$ period (16) are occupied, then assigns a new deterministic backoff value as $V(d) = x_t + x_e (16 + 4)$ and maintains the *collision-free state* by keeping the number of stations on the medium less than the deterministic backoff value.

In the next scenario, coexistence performance of the BCCA and other hybrid schemes with the DCF is examined. In this case, each medium is implemented as

half of the stations are using the BCCA or one of the hybrid schemes and the other half are using the DCF. Overall throughput performance of the coexistence situation for both saturated and unsaturated traffic under non-ideal channel condition is shown in Figure 7.15. Frame error rate is considered as 1% in this case.



a)

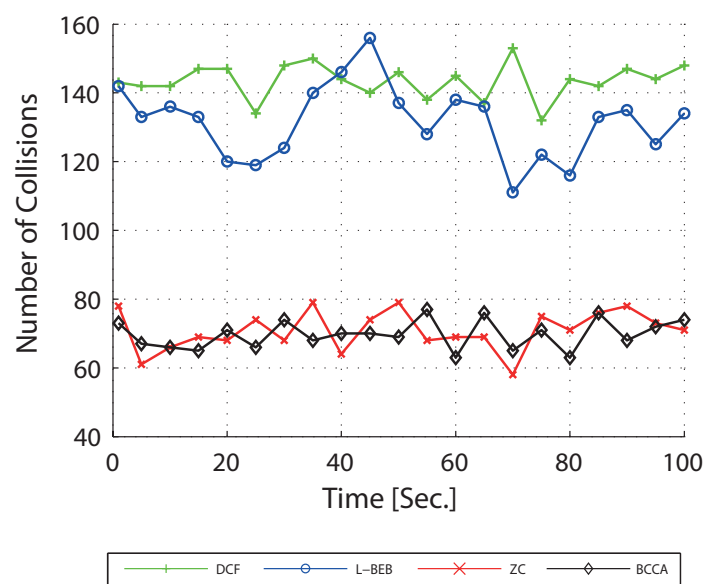


b)

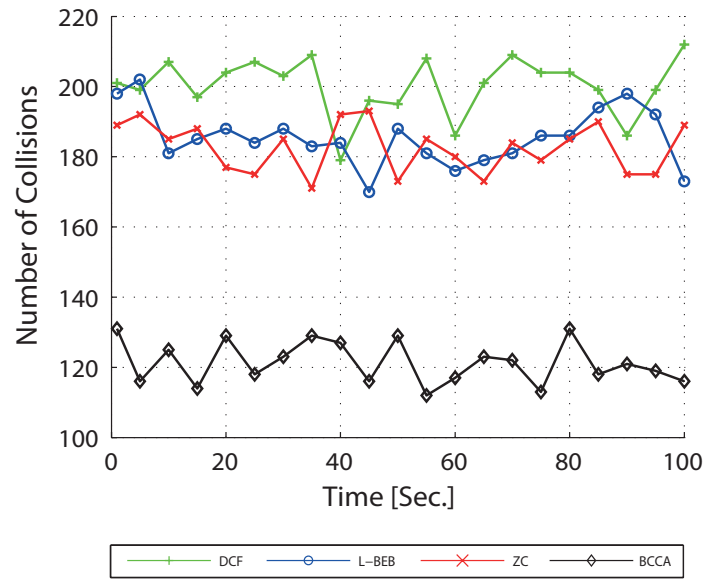
Figure 7.15: Coexistence throughput values of BCCA-DCF, ZC-DCF, L-BEB-DCF and DCF itself. a) Number of data stations is 16., b) Number of voice, data, and video stations are 6, 4, 6.

Coexistence with the DCF may cause collisions anytime in the network due to the random binary exponential backoff procedure of DCF and this randomness prevents BCCA or other hybrid schemes to converge to the *collision-free state* and hence, decreases the channel efficiency. However, as it is seen in Figure 7.15, BCCA-DCF coexistence situation mainly has higher throughput than other coexistence approaches during the simulation due to the real-time deterministic backoff adaptation as explained in the previous sections.

Figure 7.16 shows the number of collisions per second in the network for BCCA-DCF, LBEB-DCF, ZC-DCF and DCF coexistence situations. When the medium consists of saturated stations, BCCA and ZC have a similar performance and fewer collisions. It is because, in case of a collision or a frame error, they both select one of the empty slots of the last backoff period randomly for their next transmission attempt.



a)



b)

Figure 7.16: Number of collisions vs. Time for the coexistence situation. a) Number of data stations is 16, b) Number of voice, data, and video stations are 6, 4, 6.

When the medium is a combination of saturated and unsaturated stations, collision probability of hybrid schemes is almost high as DCF itself. However, *virtual frame adaptation* of BCCA prevents further collisions and lets the collision probability be as low as possible even in the coexistence situation of saturated and unsaturated stations.

7.3. Evaluation of Smart Channel Scanning Scheme

Performance of the proposed smart scanning scheme along with the *passive*, *active*, *selective*, and *unicast scanning* were analyzed by extensive simulations using the OMNET simulator and a custom-made object-oriented event-driven simulator software written in C++. Some of the parameters used to calculate formulas and implement the IEEE 802.11 MAC layer are shown in Table 7.3 and Table 7.4. We implemented our own TCP/VoIP traffic generator and evaluation tool that creates realistic data and multimedia packet streams using the parameters in Table 7.3 and Table 7.4. During the simulations, TCP frames are implemented as saturated frames. Voice frames are implemented as unsaturated frames and queued in 20ms intervals. In

order to reach saturation condition, offered load has been set greater than the maximum achievable throughput.

Table 7.3: Timing unit and values for IEEE 802.11b.

Name	Value	Unit
MinChTime	30	ms
MaxChTime	50	ms
ChSwitching	2	ms
Slot time	20	μs
Frame size	20 - 1000	Byte
Bit rate	1 - 11	Mbps

Table 7.4: IEEE 802.11b parameter values used at simulations.

Name	Length (bits)	Bit rate (Mbps)	Time (us)
DIFS	50	1	50
Preamble	192	1	192
Header	240	11	21,81
Data (TCP)	8000	11	727,27
Voice (UDP)	1280	11	116,36
SIFS	10	1	10
ACK	112	2	56

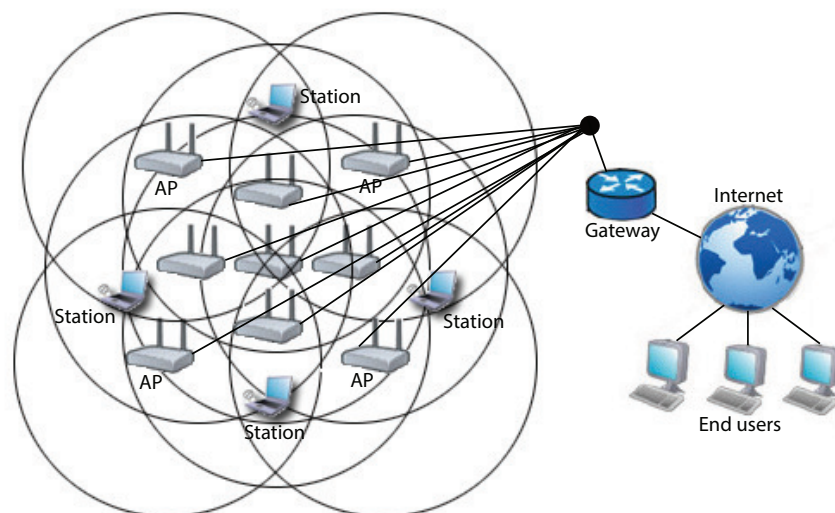
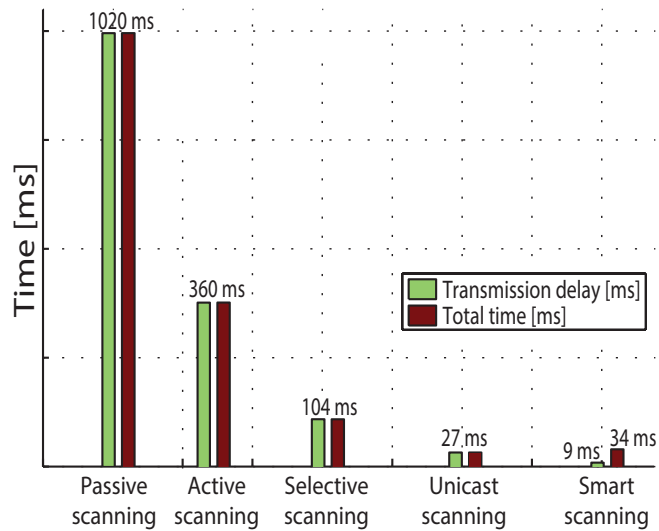
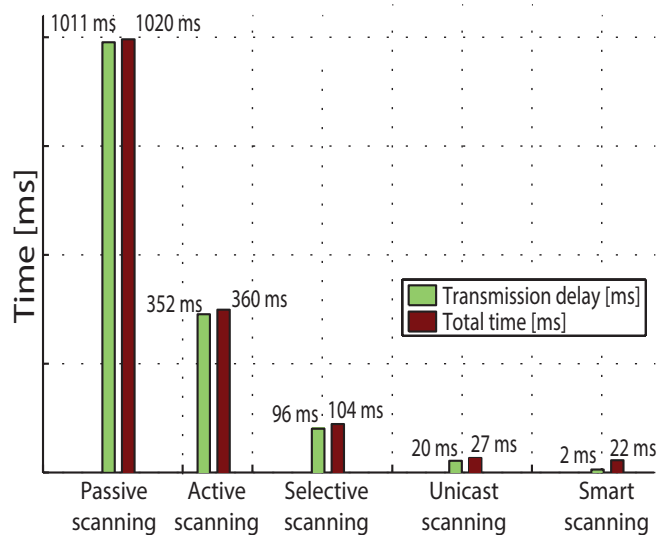


Figure 7.17: IEEE 802.11 network communication scenario.

Simulation topology is shown in Figure 7.17. It is a static environment that is composed of 10 stations (5 TCP and 5 VoIP stations) and APs connected to a wired network. The number of APs and channels are varied depending on the scenario requirements.



a)

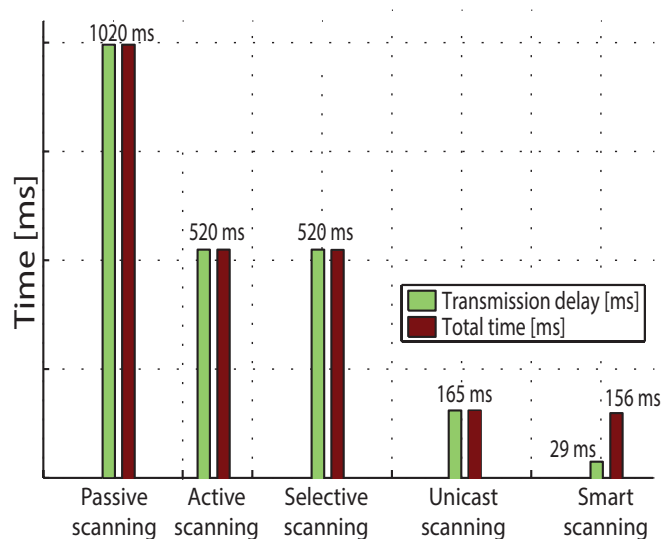


b)

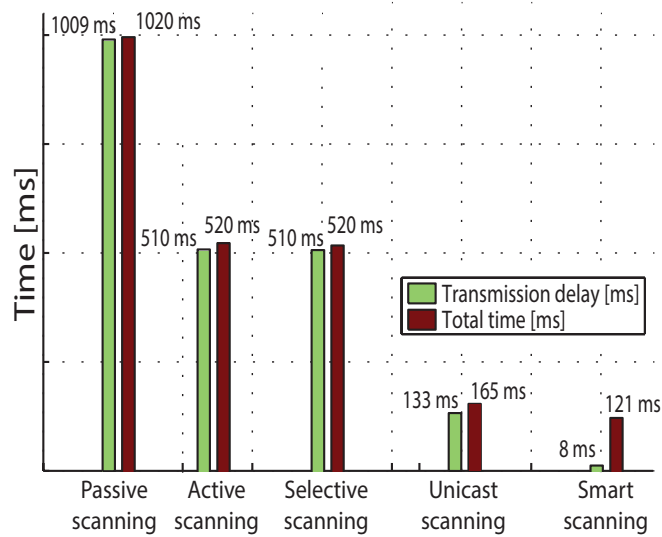
Figure 7.18: Channel scanning comparison of related schemes. a) TCP (saturated) station, b) VoIP (unsaturated) station.

In the first scenario, three APs (including the associated AP of the station) on three non-overlapping channels (channel 1, 6 and 11) are considered. Figure 7.18 shows the comparison of related schemes in terms of the total time spent for the channel scanning and the time that stations delayed their transmissions. Comparison is made by computing the average value of 10 simulation runs. The total time is the time from the beginning of scanning until the last channel is scanned. As is shown in

Figure 7.18, *passive scanning* takes the longest time (1020 ms) since stations scan 10 out of 11 channels. This duration is too long especially for VoIP stations, so that ongoing sessions will be disconnected. Compared to *passive scanning*, *active scanning* is completed within 360 ms. Additionally, *selective* and *unicast scanning* are completed in 104 and 27 ms, respectively since both schemes scan only two active channels. *Unicast scanning* is faster than the *selective scanning* since it only waits for *MinChTime* on each channel at most. The total scanning time for the proposed smart channel scanning is 34 ms (which includes 11 ms frame delivery phase as well). It provides the shortest transmission delay (9 ms) since it only scans two channels in the HCL adaptively using the procedure explained in section 6. Moreover, as is shown in Figure 5-b, transmission delay of the VoIP station is only 2 ms. It is because the smart scanning is performed after a successful frame delivery and mostly completed before a new frame is inserted into the queue of the station. Therefore, only the smart scanning scheme does not interrupt ongoing communications in this scenario. Thus, it provides QoS for delay-sensitive multimedia applications.



a)



b)

Figure 7.19: Channel scanning comparison of related schemes. a) TCP (saturated) station, b) VoIP (unsaturated) station.

In the second scenario, 11 APs operating on 11 different channels (one AP for each channel) are considered. Figure 7.19 shows the comparison of related schemes in terms of the total time spent for the channel scanning and the time that stations delayed their transmissions. This time, *active scanning* is completed in 520 ms, which is longer than the first set-up. It is because the station performing the *active scanning* has to wait for *MaxChTime* on all of 10 channels. *Selective scanning* shows the same performance with the *active scanning* since all of the channels are active in the second case. *Unicast scanning* is faster than the *selective scanning* as in the first set-up. The smart scanning is completed within 7 *scanning-phases* (156 ms) for the TCP station and within 4 *scanning-phases* (121 ms) for the VoIP station, respectively. Scanning interval T_{scan} is computed as 11 ms for the TCP station and 20ms for the VoIP station. As is shown in Figure 7.19-b, transmission delay of the VoIP station is only 8ms. Besides, the transmission delay of the TCP station (29 ms) is also reduced with the proposed scheme since the *channel-scanning* procedure is carefully scheduled together with the *frame-delivery* process in order to minimize the interruption of ongoing communication of the station.

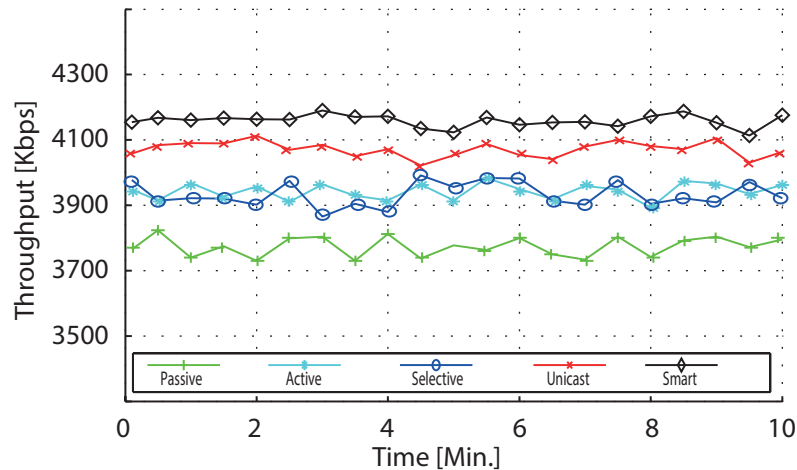


Figure 7.20: Throughput values of related schemes.

In the third scenario, 11 APs operating on 11 different channels and 10 stations (5 TCP and 5 VoIP stations) are considered as in the second scenario. We assume that periodic background scanning interval is adjusted as 10 seconds. Therefore, stations have to perform periodic background channel scanning 60 times during a simulation time of 10 minutes. Figure 7.20 shows the overall throughput values of 10 stations obtained by performing different channel scanning schemes. As is shown in Figure 7.20, the proposed smart scanning has the highest throughput since it reduces the scanning overhead and increases the efficiency of the channel utilization. Additionally, due to the high scanning overhead, *passive*, *active* and *selective scanning* schemes cause high delay variations and let stations discard some of voice frames during the simulation in order to maintain their sessions. *Unicast scanning* also degrades the quality of VoIP sessions especially after the channel scanning processes. However, delay variation is mostly around 20ms for the proposed scheme and hence, it also supports QoS with minimized communication interruptions over IEEE 802.11 WLANs.

7.4. Coexistence Evaluation of the Proposed Schemes

As it is pointed out in earlier sections, the proposed uninterrupted collision-free MAC adaptation (UCFA) allows stations to compute the deterministic value $V(d)$ analytically and to capture a slot, making use of the number of busy/empty slots in a predefined interval. In contrast, the proposed beacon based collision-free channel access scheme (BCCA) allows stations to select (reserve) a slot, making use of heuristic approach upon receiving each beacon frame. Although both schemes make use of different parameters and rely on different observations, they both serve the same purpose, which is to use the same time slot in consecutive backoff periods as in TDMA. Since selection of the transmission slot in both schemes is deterministic over a period, as soon as all stations successfully transmit and capture their slots, each of the stations periodically transmits in every $V(d)$ slot in a similar way. Therefore, both schemes have similar performance with some minor differences as shown in Figure 7.21.

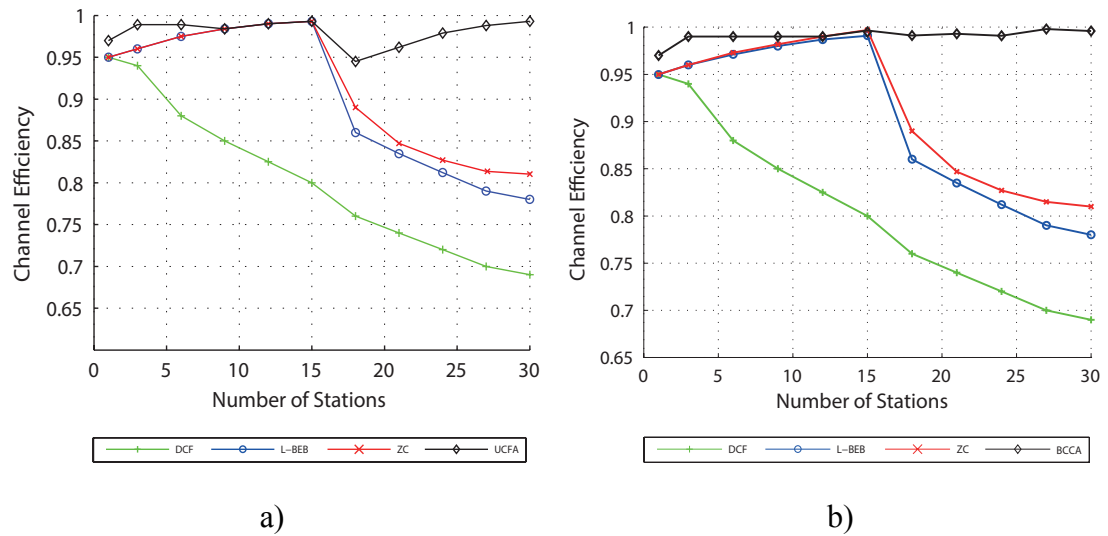
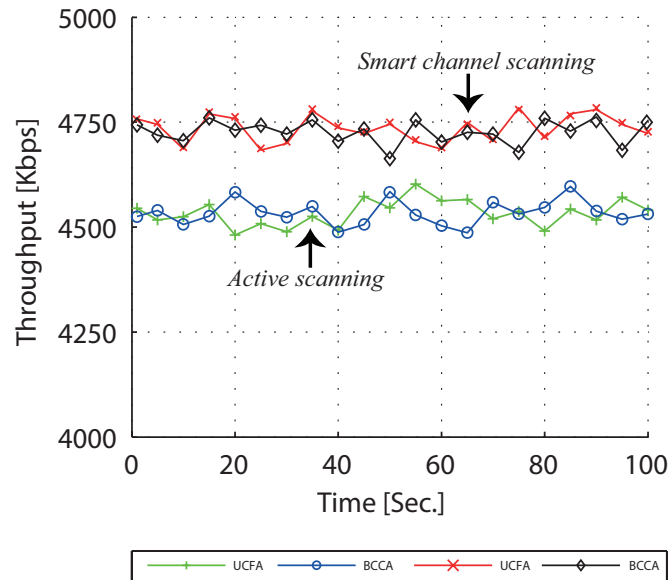


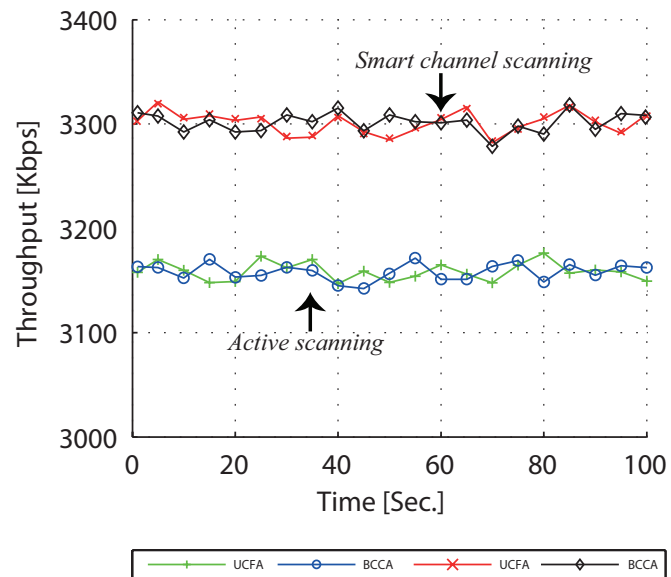
Figure 7.21: Channel efficiency vs. Number of data stations for UCFA and BCCA. a) UCFA, b) BCCA.

In order to evaluate the coexistence situation of the proposed collision-free channel access and smart selective channel scanning schemes, we have implemented a simulation topology as shown in Figure 7.17. In this scenario, 11 APs operating on 11 different channels and various numbers of stations are considered. We assume

that periodic background scanning interval is adjusted as 10 seconds. Each medium is implemented as half of the stations are using either UCFA or BCCA and the other half are using the standard DCF. Both saturated and unsaturated traffic under non-ideal channel condition are considered and the frame error rate is set as 1%.



a)



b)

Figure 7.22: Coexistence situation throughput performance of UCFA and BCCA when the smart channel scanning scheme is on and off. a) Number of data stations is 16, b) $V(d) = 16$ and the # of voice, data, video stations are 6, 4, 6.

Coexistence situation throughput performance of UCFA and BCCA when the smart channel scanning scheme is on and off can be seen in Figure 7.22. As is shown in both Figure 7.22-a and Figure 7.22-b, when the proposed channel scanning scheme is on, both UCFA and BCCA increase their throughputs (app. 4-5%) as smart channel scanning scheme reduces the channel scanning overhead and increases the efficiency of the channel utilization. As clearly seen, our proposed channel access methods and smart channel scanning scheme work in a complementary way and improve the efficiency of the channel for both *frame delivery* and *channel scanning* phases.

7.5. Energy-centric and Process-based Evaluation

Energy efficiency over WLANs is a very important topic as well. A wireless interface can be in *awake state*, *doze state* or *off state*. In the *off state*, stations consume no power. In the *doze state*, since stations cannot transmit or receive, they consume very little power. In the *awake state*, a node may be in one of three different modes, namely, *transmit*, *receive*, and *idle* modes, and consumes somewhat different power in each mode. For instance, Lucent IEEE 802.11 WaveLAN card [Stemm *et al.*, 1997] consumes 1.65 W, 1.4 W and 1.15 W in the *transmit*, *receive* and *idle* modes, respectively, in the awake state. In the doze state, WaveLAN consumes 0.045 W [Stemm *et al.*, 1997]. Clearly, a significant amount of energy is consumed even in *idle* mode. This occurs due to CSMA/CA and similarly UCFA mechanisms that require each *awake* node to listen to the channel continuously.

In our proposed schemes, wireless nodes monitor the busy/idle slots continuously by keeping a status counter in the *idle mode* in order to know when to transmit. As it is clear, this procedure is not friendly to power saving and may prevent wireless nodes to be in the *doze* (power saving) *state*. However, since UCFA prevents collisions and outperforms the standard DCF, nodes transmit more frames in unit time and thus, it may provide energy-efficient transmission especially when the medium is heavily loaded and stations are rarely asleep.

As of now, there is a tradeoff between the throughput performance and the energy efficiency in our proposed schemes. Since wireless nodes monitor the busy/idle slots continuously and never enter the *doze state*, nodes can maximize the

throughput but decrease the energy efficiency. However, there are some applicable solutions that can be more efficient than the standard DCF in terms of both throughput and energy efficiency. For example, if an algorithm that predicts an approximate time between the two consecutive $V(d)$ s and allows nodes to sleep after their successful transmissions until their next transmission attempt is implemented, nodes can wake up at the right time, transmit their frames and sleep again. This solution does not degrade the throughput performance and increases the energy efficiency dramatically. However, prediction of an approximate time between the two consecutive $V(d)$ s is not an easy job to accomplish. Another solution is to implement UCFA so that it will allow nodes to be in the *doze state* and when they wake up, nodes will chose their first backoff values randomly and proceed as the standard DCF. As soon as the transmission attempt is successful, the deterministic backoff selection will be used again. This solution can also enhance the energy efficiency dramatically but may decrease the overall throughput since collisions may occur. Yet another solution is to implement the UCFA and BCCA so that they will allow nodes to be in the *doze state* and when they wake up, nodes will wait for a $V(d)$ period (not for a beacon interval) and then select one of the empty slots of the previous $V(d)$ period. This solution may provide higher throughput than the former solution but may increase the delay. All the solutions mentioned above are applicable and can increase the energy efficiency of the proposed schemes.

Additionally, in our proposed methods UCFA and BCCA, transmitting frames in collision-free state over a communication network requires additional computational efforts. In addition to the computational efforts of the standard DCF, in UCFA, stations first computes the probability of collision and the expected number of stations, respectively upon receiving each beacon frame using the equation 4.3, 4.6 and the Extended Kalan Filter given through the equation 4.8 – 4.13. Afterwards, stations compute the deterministic backoff value $V(d)$ using the equation 4.7. This processing overhead however let stations enter the collision-free state and outperform the standard DCF in terms of delay, packet loss rate and overall throughput. Besides, stations run these additional computations once in every beacon frame reception. Therefore, processing overhead occurs only after the beacon frame receptions. Since stations can compute the required equations in the DIFS, SIFS or

other unused intervals, they will not delay their transmissions due to the processing overhead caused by the UCFA procedure.

In addition to the computational efforts of the standard DCF, in BCCA, stations compute the number of approaching slots once in every $V(d)$ period. As in UCFA, stations implemented with BCCA can compute the required equations in the DIFS, SIFS or other unused intervals, and hence they will not delay their transmissions due to the processing overhead caused by the BCCA procedure. This processing overhead however let stations enter the collision-free state and outperform the standard DCF in terms of delay, packet loss rate and overall throughput.

8. CONCLUSION

In this thesis, we first propose an adaptive uninterrupted collision-free MAC adaptation scheme called UCFA to improve the performance of the DCF/EDCA mechanisms. The idea is based on the local estimation of channel's contention level by measuring the duration of busy and idle periods observed on the channel at each station. In order to prevent the collision probability, the proposed scheme has a deterministic backoff value according to the channel contention level. The proposed scheme allows a station to reuse a time slot in consecutive backoff cycles by deterministically setting its backoff counter upon successful frame transmissions.

Second, we propose an adaptive beacon-based collision-free MAC adaptation called BCCA. The proposed scheme makes use of beacon frames sent periodically by Access Point (AP), lets stations enter the collision-free state and reduces the number of idle slots regardless of the number of stations and their traffic load (saturated or unsaturated) on the medium. In order to prevent collisions, this scheme also makes use of a deterministic backoff value according to the channel contention level. The key idea of the proposed scheme is to carefully control the backoff process and achieve additional resource reservation without any extra cost.

UCFA and BCCA have the following characteristics compared to the random backoff and the CSMA-TDMA hybrid approaches. They exceed the maximum theoretical performance of the CSMA/CA. They can provide an uninterrupted collision-free medium access regardless of the traffic condition (saturated or unsaturated) and the number of stations on the medium. They can fairly coexist with the legacy CSMA/CA and work in a distributed fashion. They require some additional computational efforts. However, they can be implemented without any changes on the current IEEE 802.11 standard. Finally, they are robust against channel errors and achieve much better network performance at near zero cost.

In addition to the proposed collision-free channel access approaches, a smart channel scanning method is also proposed within the scope of this thesis. The proposed method minimizes the ongoing communication interruptions in a distributed manner by reducing both the size of the set of channels to be scanned and the scanning time on each channel. In the proposed smart channel scanning method, channel scanning is performed after a successful frame delivery and it is scheduled

with multiple smart scanning intervals. Simulation results show that our proposed scheme significantly reduces the scanning overhead, unnecessary handovers, and force disconnections. Moreover, it does not require any modification on the current IEEE 802.11 WLAN standard and hence, it can be easily integrated into the existing mobile devices.

REFERENCES

- Abd-Elnaby M., Rizk M. R. M., Dessouky M. I., El-Dolil S. A., (2011), "Efficient contention-based MAC protocol using adaptive fuzzy controlled sliding backoff interval for wireless networks", *Computers & Electrical Engineering*, 37 (1), 115-125.
- Acharya P. A. K., Sharma A., Belding E. M., Almeroth K. C., Papagiannaki K., (2010), "Rate Adaptation in Congested Wireless Networks through Real-Time Measurements," *IEEE Transactions on Mobile Computing*, 9 (11), 1535 – 1550.
- Alonso L., Agustí R., (2006), "Optimization of wireless communication systems using cross-layer information", *Signal Processing Journal*, 86 (8), 1755-1772.
- Barcelo J., Bellalta B., Cano C., and Oliver M., (2008), "Learning-BEB: Avoiding Collisions in WLAN", *Eunice Summer School, Brest, France*, 8–10 September.
- Barcelo J., Bellalta B., Cano C., Sfairopoulou A., and Oliver M., (2010), "Dynamic Parameter Adjustment in CSMA/ECA", *MACOM, Barcelona, Spain*, 13-14 September.
- Barcelo J., Bellalta B., Cano C., Sfairopoulou A., Oliver M., and Zuidweg J., (2009), "Traffic Prioritization for Carrience Sense Multiple Access with Enhanced Collision Avoidance", *IEEE ICC (MACOM), Dresden, Germany*, June.
- Barcelo J., Bellalta B., Sfairopoulou A., Cano C., and Oliver M., (2009), "CSMA with Enhanced Collision Avoidance: a Performance Assessment", *IEEE VTC-Spring, Barcelona, Spain*, April.
- Barcelo J., Bellalta B., Sfairopoulou A., Cano C., and Oliver M., (2009), "Carrier Sense Multiple Access with Enhanced Collision Avoidance: a Performance Analysis", *ACM IWCMC, Leipzig, Germany*, 21-24 June.
- Bianchi G., (2000), "Performance analysis of the IEEE 802.11 distributed coordination function", *IEEE Journal on Selected Areas in Communications*, 18, 535–547.
- Bianchi G., and Tinnirello I., (2003), "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network", *IEEE Infocom, San Francisco, CA*, April.
- Calì F., Conti M., Gregori E., (2000), "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit", *IEEE/ACM Transactions on Networking*, 8 (6), 785-799.

Chatzimisios P., Boucouvalas A. C., (2005), "Packet delay analysis of the advanced infrared (AIr) CSMA/CA MAC protocol in optical wireless LANs", *International Journal of Communication Systems*, 18 (3), 307-331.

Chatzimisios P., Vitsas V., Boucouvalas A. C., and Tsoulfa M., (2007), "Achieving performance enhancement in IEEE 802.11 WLANs by using the DIDD backoff mechanism", *International Journal Of Telecommunication Systems*, 20 (1), 23-41.

Chih-Heng K., Chih-Cheng W., Kawuu W. L., Jen-Wen D., (2011), "A smart exponential-threshold-linear backoff mechanism for IEEE 802.11 WLANs", *International Journal Of Telecommunication Systems*, 24 (8), 1033-1048.

Chintala V. M., Zeng Q. A., (2007), "Novel MAC Layer Handoff Schemes for IEEE 802.11 Wireless LANs", *IEEE WCNC, Hong Kong*, 11-15 March.

Dong-Wan K., Sang-Jo Y., (2009), "An Efficient Scanning Group and Order Decision Method Using Neighbor Network Information in Wireless LAN", *ISCIT, Icheon*, 28-30 September.

Ekici O., Yongacoglu A., (2008), "IEEE 802.11a Throughput Performance with Hidden Nodes", *IEEE Communications Letters*, 12 (6), 465-467.

Fang M., Malone D., Duffy K. R., Leith D. J., (2013), "Decentralised Learning MACs for Collision-free Access in WLANs", *Wireless Networks*, 19 (1), 83-98.

Gyung-Ho H., Dong-Ho C., (2005), "New access scheme for VoIP packets in IEEE 802.11e wireless LANs", *IEEE Communications Letters*, 9 (7), 667-669.

Haas Z. J., Deng J., (2003), "On optimizing the backoff interval for random access schemes", *IEEE Transactions on Communications*, 51 (12), 2081-2090.

Haitao W., Fan Z., Qian Z., Zhisheng N., (2006), "Analysis of IEEE 802.11 DCF with hidden terminals", *IEEE GLOBECOM*, November.

Heusse M., Rousseau F., Guillier R., Duda A., (2005), "Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs", *Proc. IEEE SIGCOMM, New York, USA*, October.

IEEE Std 802.11, (1999), "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications".

IEEE Std 802.11e, Part 11, (2005), "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications", Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements.

IEEE Std. 802.21, (2008), “Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Service.” IEEE 802.21/D11.0, May.

IEEE802.11 WaveLAN PC Card - User’s Guide, p.A-1.

Igor K., Young-Tak K., (2011), “Prediction-Based Smart Channel Scanning with Minimized Service Disruption for IEEE 802.11e WLAN”, IEEE Transactions on Consumer Electronics, 57 (2), 386-394.

ITU-T Recommendation G.107, (2000), “The E-model, A Computational Model for Use in Transmission Planning”, May.

Jain R., Durresi A., and Babic G., (1999), “Throughput fairness index: an explanation”, in Proc. ATM Forum/99-0045, February.

Jyh-Cheng C., Tuan-Che C., Tao Z., Eric V. D. B., (2006), “Effective AP selection and Load Balancing in IEEE 802.11 Wireless LANs”, IEEE Globecom, San Francisco, CA, November.

Kassar M., Kervella B., Pujolle G., (2008), “An Overview of vertical handover decision strategies in heterogeneous wireless networks,” Computer Communications, 31 (10), 2607 – 2620.

Kim S. W., Kim B. S., Fang Y., (2010), “Network allocation vector (NAV)-based opportunistic prescanning process for WLANs”, Journal of Electronics Letters, 46 (24), 1630-1632.

Klaus D., C’assio B. R., Jarkko K., (2011), “On efficient discovery of next generation local area networks”, IEEE WCNC, Quintana-Roo, Mexico, 28-31 March.

Kwon Y., Fang Y., Latchman H., (2004), “Design of MAC Protocols With Fast Collision Resolution for Wireless Local Area Networks”, IEEE Transactions on Wireless Communications, 3 (3), 793-807.

Lee J., Walrand J., (2008), “Design and analysis of an asynchronous zero collision MAC protocol”, Arxiv preprint arXiv:0806.3542.

Li B., Battiti R., Fang Y., (2007), “Achieving optimal performance by using IEEE 802.11 MAC protocol with service differentiation enhancements”, IEEE Transactions on Vehicular Technology, 56 (3), 1374-1387.

Malone D., Clifford P., Leith D. J., (2007), “MAC layer channel quality measurement in 802.11”, IEEE Communication Letters, 11 (2), 143-145.

- Ni Q., Romdhani L., Turletti T., (2004), "A Survey of QoS Enhancements for IEEE 802.11 Wireless LAN", *Journal of Wireless Communications and Mobile Computing*, 4 (5), 547-566.
- Papachristou C., Pavlidou F. N., (2002), "Collision-free operation in ad hoc carrier sense multiple access wireless networks", *IEEE Communications Letters*, 6 (8), 352-354.
- Raghavendra R., Belding E. M., Papagiannaki K., Almeroth K. C., (2010), "Unwanted Link Layer Traffic in Large IEEE 802.11 Wireless Networks," *IEEE Transactions on Mobile Computing*, 9 (9), 1212–1225.
- Ramani, I., Savage, S., (2005), "SyncScan: practical fast handoff for 802.11 infrastructure network," *IEEE Infocom*, Miami, 13 -17 March.
- Sakurai T., Vu H. L., (2007), "MAC Access Delay of IEEE 802.11 DCF", *IEEE Transactions on Wireless Communications*, 6 (5), 1702-1710.
- Shafer I., Chang M., (2010), "Movement Detection for Power-Efficient Smartphone WLAN Localization", In *Proc. of the 13th ACM int'l conf. on Modeling, analysis, and simulation of wireless and mobile systems*, Bodrum, Turkey, 17-21 October.
- Shin M., Mishra A., Arbaugh W. A., (2004), "Improving the Latency of 802.11 Hand-offs using Neighbor Graphs," *Processing of the ACM MobiSys Conference*, Boston, MA, USA, 70-83, June.
- Sooyong L., Myungchul K., Sungwon K., Kyunghye L. IlGu J., (2012), "Smart Scanning for Mobile Devices in WLANs" *IEEE ICC-2012*, Ottawa, Canada, 10-15 June.
- Stemm M. Randy H. K., (1997), "Measuring and reducing energy consumption of network interfaces in hand-held devices," *IEICE Transaction on Communications*, special Issue on Mobile Computing, 80 (8), 1125–1131.
- Sun-Myeng K., Young-Jong C., (2006), "A distributed collision resolution scheme for improving the performance in wireless LANs", *Computer Networks*, 50 (3), 289-300.
- Taniuchi K., Ohba Y., Fajardo V., (2009), "IEEE 802.21: Media Independent Handover: Features, Applicability and Realization," *IEEE Communications Magazine*, 47 (1), 112–120.
- Tinnirello I., (2008), "Kalman Filter estimation of the contention dynamics in error-prone IEEE 802.11 networks", *ISCCSP 2008*, Malta, 12-14 March.

Toledo A. L., Vercauteren T., Wang X., (2006), "Adaptive Optimization of IEEE 802.11 DCF Based on Bayesian Estimation of the Number of Competing Terminals", IEEE Transactions On Mobile Computing, 5 (9), 1283-1296.

Tuysuz M. F., Mantar H. A., (2010), "A Cross Layer QoS Algorithm to Improve Wireless Link Throughput and Voice Quality over Multi-rate WLANs", IWCMC-2010, Caen, France, 26 - 29 June.

Tuysuz M. F., Mantar H. A., (2010), "Access Point Selection and Reducing the Handoff Latency to Support VoIP Traffic", ICCES-2010, Cairo, Egypt, 30 November.

Tuysuz M. F., Mantar H. A., (2010), "Access Point Selection for Improving the Voice Quality and Overall Throughput in Wireless LANs", SoftCom-2010, Split, Croatia, 23 - 25 September.

Tuysuz M. F., Mantar H. A., (2010), "An Adaptive Codec and Frame Size Modification Based QoS Algorithm over Multi-rate WLANs", WIMO-2010, Ankara, Turkey, 24 - 26 June.

Tuysuz M. F., Mantar H. A., (2010), "Minimizing the Effects of Multi-rate WLANs by Adapting Link Adaptation and Call Admission Interfaces", NDT-2010, Prague, Czech Republic, 3 - 5 July.

Tuysuz M. F., Mantar H. A., (2011), "Adaptive Channel Access Tuning and QoS Support for Multimedia Applications over IEEE 802.11e WLANs", ICIMU-2011, Kuala Lumpur, Malaysia, 14 - 16 November.

Tuysuz M. F., Mantar H. A., (2013), "A Beacon-based Collision-Free Channel Access Scheme for IEEE 802.11 WLANs", To appear on Wireless Personal Communication.

Tuysuz M. F., Mantar H. A., (2013), "A Novel Beacon-based Collision-Free Channel Access Mechanism over IEEE 802.11 WLANs", IEEE WCNC 2013, Shangai, China, April.

Tuysuz M. F., Mantar H. A., (2013), "An Uninterrupted Collision-Free Channel Access Scheme over IEEE 802.11 WLANs", IEEE WCNC 2013, Shangai, China, April.

Tuysuz M. F., Mantar H. A., (2013), "Exploiting the Channel Using Uninterrupted Collision-Free MAC Adaptation over IEEE 802.11 WLANs", To appear on Wireless Communications and Mobile Computing.

Tuysuz M. F., Mantar H. A., (2013), "Smart Channel Scanning with Minimized Communication Interruptions over IEEE 802.11 WLANs", IEEE PIMRC-13, London, England, 8 - 11 September.

Wang F., Ghosh A., Sankaran C., Fleming P. J., Hsieh F., Benes S. J., (2008), "Mobile WiMAX Systems: Performance and Evolution," IEEE Communications Magazine, 46 (10), 41–49.

Wei L., Huan J., Xinbing W. Mohsen G., (2011), "A novel IEEE 802.11-based MAC protocol supporting cooperative communications", International Journal of Telecommunication Systems, 24 (11), 1480–1495.

Xiao Y., (2005), "Performance analysis of priority schemes for IEEE 802.11 and IEEE 802.11e wireless LANs", IEEE Transactions on Wireless Communications, 4 (4), 1506-1515.

Yamada, H., Morikawa H., Aoyama T., (2003), "Decentralized Control Mechanism Suppressing Delay Fluctuation in Wireless LANs", IEEE VTC-fall, Orlando, USA, 6-9 October.

Yong H., Jie S., Ruixi Y., Weibo G., (2009), "Semi-Random Backoff: Towards Resource Reservation for Channel Access in 802.11e Wireless LANs", ICNP'09, Princeton, NJ, 13-16 October.

Yong H., Jie S., Ruixi Y., Xiaojun M., Athanasios V. V., Weibo G., (2013), "Semi-Random Backoff: Towards Resource Reservation for Channel Access in Wireless LANs", To appear on IEEE/ACM Transactions on Networking.

Yu X., She Y., Li B., Lei L., Lang S., (2009), "Predictive Channel Scanning over WLAN," WiCom, Beijing, China, 24-26 September.

Ziouva E., Antonakopoulos T., (2002), "CSMA/CA performance under high traffic conditions: throughput and delay analysis", Computer Communications, 25 (3), 313-321.

RESUMÉ

Mehmet Fatih Tüysüz holds the B.Sc. degrees from İnönü University Department of Electric and Electronic Engineering, Malatya/Turkey. He holds the M.Sc. degree from Dokuz Eylül University Department of Electric and Electronic Engineering, Izmir/Turkey. During his M.Sc. thesis, he worked on the “Quality of Service Enhancement of VoIP applications over wireless networks” and published several papers in this area. He joined Gebze Institute of Technology Computer Engineering department in September 2008 as a Ph.D. student. He has been working as a research assistant in the same department. His interest includes VoIP, Wireless QoS, energy-aware communications and energy optimization in wireless networks.

APPENDIX

Algorithm 2: Uninterrupted Collision-Free MAC Adaptation (UCFA)

```
1: while (there is a frame to transmit)
2:    $t_a = 0$ 
3:   while ( $t_a < m_t$ )
4:     while (station hasn't sent any frame yet AND has no beacon)
5:       wait for the new beacon frame
6:     end
7:     if (new beacon frame is received) then
8:       estimate the total number of stations  $x_t$ 
9:        $V(d) = V_{dconvergence}(x_t + x_n)$  // calculate the  $V(d)$  according to  $x_t$ 
10:      if (station has not sent any frame yet) then
11:        analyze previous  $V(d)$  periods to prioritize empty slots
12:        if (there are high priority slots) then
13:          backoff = random(one of the high priority slots)
14:        else
15:          backoff = random(one of the low priority slots)
16:        while (backoff > 0)
17:          backoff = backoff - 1 // wait for one slot
18:        end
19:        attempt transmission
20:        if (transmission is successful) then
21:          backoff =  $V(d)$ 
22:          break
23:        else
24:           $t_a = t_a + 1$ 
25:          if (station was in deterministic-state in the prev.  $V(d)$ ) then
26:            backoff =  $V(d)$ 
27:          else
28:            backoff = random(one of the empty slots)
29:          end
30:           $V(d) = 2 * V(d)$ 
31:          backoff =  $V(d)$ 
32:        end
33:      while (there is no frame to transmit)
34:        initialize the virtual frame  $V_f$  to the last successful trans. slot
35:         $V_f = V(d)$ 
36:        while ( $V_f > 0$ )
37:           $V_f = V_f - 1$  // wait for one slot
38:          if (there is a frame to transmit) then
39:            backoff =  $V_f$ 
40:            if (the slot was captured by a sta. in the prev.  $V(d)$ ) then
41:              analyze previous  $V(d)$  periods to prioritize empty slots
42:              if (there are high priority slots) then
43:                backoff = random(one of the high priority slots)
44:              else
45:                backoff = random(one of the low priority slots)
46:              go to 2
47:            if ( $V_f == 0$ ) then
48:               $V_f = V(d)$ 
49:            end
50:          end
```

Algorithm 3: Beacon-based Collision-free Channel Access (BCCA)

```

1:  $V(d)$ ,  $a_p$  and  $m_c$  initializations
2: while (there is a frame to transmit)
3:   if (the station has not sent any frame yet) then
4:     wait for one  $V(d)$  period // 16 slots for the initialization
5:     maintain the status of the last  $V(d)$  slots
6:     if (there are empty slots in between  $V(d)$  slots) then
7:       backoff = random(one of empty slots of the last  $V(d)$  slots)
8:     else
9:        $V(d) = 2 * V(d)$ 
10:    go to 4
11:  while (backoff > 0)
12:    maintain the status of the last  $V(d)$  slots
13:    if (there is no empty slots in  $V(d)$  slots) then
14:       $V(d) = x_i + x_e$  //  $1 \leq x_e \leq 4$ 
15:      if (other stations have a repetitive slot occupancy in a  $V(d)$ ) then
16:         $a_p = a_p + 1$ 
17:        if ( $a_p == m_c$ ) then
18:           $V(d) = V(d)/2$ 
19:           $a_p = 0$ 
20:        if (new beacon is received) then
21:          if (there is a station has a lower backoff than my station) then
22:            if (there are empty slots between that station and mine) then
23:              backoff = backoffnearest sta. +  $x_{neighbor} + 1$ 
24:            if (all stations has captured the slots side-by-side) then
25:               $V(d) = x_i + x_e$ 
26:              backoff = backoff - 1
27:            attempt transmission
28:            if (transmission is successful) then
29:              backoff =  $V(d)$ 
30:            else
31:              backoff = random(one of empty slots of the last  $V(d)$  slots)
32:          while (there is no frame to transmit)
33:            assign the virtual frame  $V_f$  to the last successful trans. slot
34:             $V_f = V(d)$ 
35:            while ( $V_f > 0$ )
36:               $V_f = V_f - 1$  // wait for one slot
37:            if (there is a frame to transmit) then
38:              backoff =  $V_f$ 
39:            go to 2
40:          if ( $V_f == 0$ ) then
41:             $V_f = V(d)$ 

```
