Faculty of Information & Communication Technologies
Swinburne University of Technology
Centre for Advanced Internet Architectures
Melbourne, Australia

Analysis, Limitations and Improvements of Voice over IP in IEEE 802.11 WLAN

Kewin Oliver Stöckigt
M.Sc., B.Sc.

A thesis submitted in total fulfillment of the requirements of the degree of
Doctor of Philosophy

May 2012

Abstract

Wireless voice over IP (VoIP) is an important emerging service in telecommunications due to its potential for replacing todays' cell phone communication wherever wireless local area networks (WLANs) are deployed. However, the number of VoIP calls that can be supported in the widely deployed IEEE 802.11 WLAN is limited and therefore may hinder the success of wireless VoIP as a viable alternative. In this thesis we investigate the limited VoIP capacity and provide new insight and solutions to improve the number of simultaneous VoIP calls that can be maintained with an acceptable level of voice call quality.

To improve the VoIP capacity we first discuss a solution based on a quality of service (QoS) mechanism of the IEEE 802.11 protocol. In particular we utilize a transmission opportunity (TXOP) parameter of the medium access control (MAC) as a simple solution to increase the number of VoIP calls. We develop a detailed analytical model to show that a significant increase ($\approx 100\%$) in call capacity can be achieved. We also discuss the implications of this solution and provide ideal parameter settings to maximize the number of concurrent calls without affecting the overall system performance.

Using our analytical model, we study the impact of the buffer size on VoIP capacity. We show that there exists an optimal buffer size that maximizes the number of VoIP calls, and we show that further increasing the buffer beyond this ideal value will not result in an increase in call capacity. In particular, we show that VoIP capacity is independent of the buffer size, given the minimum optimal buffer, and that this finding also holds in conjunction with the previous solution to increase the VoIP capacity. Based on this finding, we develop a closed-form expression for the maximum number of VoIP calls as a function of the TXOP parameter. Using the closed-form expression we then propose a novel VoIP capacity approximation equation. This simple yet accurate approximation formula allows us to provide some further insight into VoIP capacity in WLAN and its limits.

We then propose a novel dynamic codec with priority (DCwP) scheme that exploits a tradeoff between call quality and traffic priority to increase call capacity while providing a high level of voice call quality. In this scheme, users are encouraged to switch to a lower quality voice codec during periods of high contention, thus enabling them to maintain the call. To compensate for the reduction in call quality, and to give an incentive to the user to adjust the voice codec, a higher priority is given at the AP for voice calls originating from these users, thus providing a high throughput and a higher than average call quality. To show the benefits of this scheme, we develop a detailed analytical model, and show that depending on the voice codec setting that a VoIP capacity gain of between 16% and almost 300% can be achieved with an above average call quality as indicated by results obtained using the ITU E-model. Our analytical model also allows us to provide further insight into a multi-queue, multi-traffic environment, in particular the impact of internal collisions.

Finally, we investigate the impact of TCP traffic on VoIP capacity in an IEEE 802.11 WLAN. Using ns-2 simulation we study the impact of TCP traffic on the call capacity when the channel access is controlled using the distributed coordination function (DCF) or enhanced distributed channel access (EDCA) is used. We confirm that the TCP flow direction is an important parameters to consider, irrespective of the channel access function. We then show that the solution based on the TXOP parameter can provide benefits to VoIP and TCP traffic. In particular, we show that using the TXOP parameter yields a high call capacity while maximizing the TCP throughput. However, we also show that even though the TXOP solutions increase the number of calls and the TCP throughput, our proposed DCwP scheme can provide superior results in terms of VoIP capacity and TCP throughput for different scenarios.

Acknowledgement

I would like to thank my supervisors A/Prof. Dr. Hai L. Vu and Dr. Philip Branch for their support for the duration of my PhD. Hai and Phil took me under their wings in early 2008 and provided a friendly and encouraging environment. Their support, encouragement and advice in particular during times when I could not see light at the end of the tunnel is very much appreciated.

I also want to express by deepest gratitude to my wife Naomi. Without her love and compassion, this journey would have been impossible. She was always there for me and supported me all the way. Thank you my love.

I also would like to thank my brother Christian and my parents. Their love, support and encouragement over the years have been invaluable.

Last but not least I would like to thank the entire CAIA lab for the great environment and the fun conversations. In particular I would like to thank Sebastian Zander, which whom I could continue to speak German. I also want to thank Md Imrul Hassan with whom I shared countless cups of coffee during the daily ”I have to check my car-coffee-walk” and as many conversations and discussions about photography.

Dedicated to my wife Naomi, my brother Christian and
my parents for their love and support.

Declaration

To the best of my knowledge and belief, this thesis contains no material previously published or written by any other person, except where due reference is made in the text of the thesis. This thesis has not been submitted previously, in whole or in part, to qualify for any other degree or diploma. The content of the thesis is the result of work, which has been carried out since the beginning of my candidature in March 2008.

Kewin Oliver Stöckigt
3 May 2012

Publications

Refereed Journal articles

Stoeckigt, K.O., Vu, H.L.; *VoIP Capacity - Analysis, Improvements and Limits in IEEE 802.11 Wireless LAN*, IEEE Transactions on Vehicular Technology, Vol. 59, Issue 9, 2010, pp 4553-4563

Stoeckigt, K.O.; Vu, H.L.; Branch, P.; *The Dynamic Codec with Priority Scheme for VoIP in IEEE 802.11 WLANs*, (under preparation)

Peer reviewed conference papers

Stoeckigt, K.O., Vu, H.L.; *VoIP capacity in IEEE 802.11 WLAN*, IEEE 34th Conference on Local Computer Networks (LCN), Zurich, Switzerland, 2009, pp. 116-123

Stoeckigt, K.O., Vu, H.L., Branch, P.; *Dynamic Codec with Priority for Voice over IP in WLAN*, Proceedings of the 2nd annual ACM conference on Multimedia systems (MMSys'11), San Jose, USA, 2011, pp. 199-210

Contents

List of Figures

List of Tables

List of Abbreviations

AC	Access category
ACK	Acknowledgement
AIFS	Arbitrary inter-frame space
AP	Access point
BDP	Bandwidth delay product
BEB	Binary exponential backoff
CAC	Call admission control
CBR	Constant bit rate
CFP	Contention free period
CNG	Comfort noise generation
CoS	Class of service
CP	Contention period
CTS	Clear to sent
CWND	Congestion window
DBR	Dynamic bit rate
DCF	Distributed coordination function
DIFS	Distributed inter-frame space
DTX	Discontinued transmission
E2E	End-to-end
EDCA	Enhanced distributed coordination function
ENUM	E.164 number mapping
FEC	Forward error correction
GSM	Global System for Mobile communication
HCCA	HCA controlled channel access
HCF	Hybrid coordination function
IFS	Inter-frame space
IP	Internet Protocol
LAN	Local area network

MAC	Medium access control
MOS	Mean opinion score
NAT	Network address translation
P2P	Peer-to-peer
PC	Point coordinator
PCF	Point Coordination Function
PCS	Physical coding sublayer
PDV	Packet delay variation
PHY	Physical layer
PIFS	PCF inter-frame space
PLC	Packet loss concealment
PLCP	Physical layer convergence protocol
PSTN	Public switched telephone network
QoE	Quality of experience
QoS	Quality of service
RTP	Real-time transport protocol
RTS	Request to sent
SIFS	Short inter-frame space
SIP	Session Initiation Protocol
SNR	Signal to noise ratio
TCP	Transmission control protocol
TXOP	Transmission opportunity
UDP	User datagram protocol
VAD	Voice activity detection
VBR	Variable bit rate
VoIP	Voice over IP
WLAN	Wireless local area network
WME	Wireless multimedia extension

List of Notations

$\lambda\epsilon$	Packet arrival rate (constant bit rate flows)
$\tilde{\lambda}$	Packet arrival rate (variable bit rate)
$\mu\epsilon$	Packet service rate
$\rho\epsilon$	Queue utilization and/or probability of having a packet to send
$c\epsilon$	Collision probability
$\tau\epsilon$	Attempt probability of transmitting a packet
$\eta\epsilon$	$TXOPLimit\epsilon$duration in packets per channel access
T_c	Collision time
T_s	Successful transmission time
T_s^*	Successful transmission time of $\eta\epsilon - \not{1}$ packets in the TXOP-frame
T_{s^*}	Successful transmission time of priority packets
T_p	Transmission time of the (raw) packet (payload + header)
T_{p^*}	Transmission time of priority (raw) packet (payload + header)
T_{ACK}	Transmission time of the MAC-ACK frame
$T_{ACK_{TO}}$	MAC-ACK timeout period
T_{AIFS}	Duration of the $AIFS\epsilon$
T_{DIFS}	Duration of the $DIFS\epsilon$
T_{SIFS}	Duration of the $SIFS\epsilon$
$\bar{t}\epsilon$	Average collision time
$\overline{w}\epsilon$	Average backoff window
$W\epsilon$	Backoff window
$m\epsilon$	Maximum backoff stage
$R\epsilon$	Retry limit
$\sigma\epsilon$	Backoff slot length
$N\epsilon$	Number of stations
ϵ	Compensation for the different backoff process in EDCA
$p\epsilon$	Packet loss probability
$C\epsilon$	Number of VoIP calls

$\hat{C}\epsilon$	Number of VoIP calls with the DCwP scheme
$K\epsilon$	Queue length in packets
$\kappa\epsilon$	Packet loss threshold
$d\epsilon$	Network delay
CW_{min}	Minimum contention window
CW_{max}	Maximum contention window
$L\epsilon$	Packet payload size
$\psi\epsilon$	VoIP activity ratio
$\overline{f}(\eta)$	Number of VoIP calls with $\eta\epsilon$(VoIP capacity approximation)
V_d	Default access queue
V_p	Priority access queue
$\delta\epsilon$	Internal collision probability
$\gamma\epsilon$	Transmission probability of a standard packet
R_Q	$R\epsilon - \not{v}$alue of the ITU E-model
$M\epsilon$	Number of TCP nodes
M_u	Number of uplink TCP senders
M_d	Number of downlink TCP sender

1
Introduction

In recent years, wireless local area networks (WLANs) based on the IEEE 802.11 protocol [4] have been widely deployed and are now available almost everywhere. Many places such as cafes, restaurants, airport lounges and shopping centers provide free wireless access. Enhancements to the wireless protocol and the reduction of costs of wireless technology over the past decade has led to an integration of wireless capabilities into mobile devices such as laptops, PDAs[1] and cell phones. In particular the widespread use of *smart-phones* sees an increasing demand for wireless access everywhere. For example, it is predicted that the mobile data traffic between 2010 and 2015 will increase by 92% annually and that mobile data will consume in excess of 3 exabyte (3×10^{12}MB) per month in 2014 as shown in Fig. 1.1 [5].

The success of wireless networks is closely coupled to the standardization efforts of the *Institute of Electrical and Electronics Engineers* (IEEE[2]) and the *European Telecommunications Standards Institute* (ETSI[3]). Both organizations established working groups in the early 1990s to develop a standardized wireless network protocol. In 1996 ETSI released the HiperLAN/1 (HIgh PErformance Radio LAN) [6–10] type 1 protocol, and HiperLAN/2 was ratified in

[1] Personal Digital Assistant
[2] http://www.ieee.org
[3] http://www.etsi.org

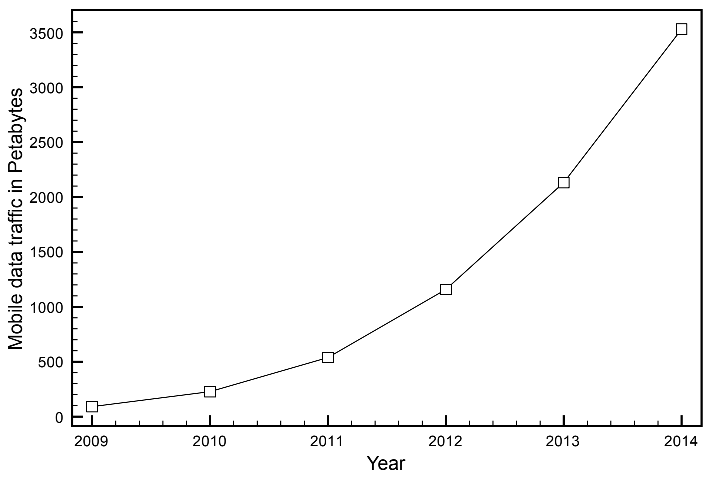

Figure 1.1: Predicted increase in mobile data (petabytes) between 2009 and 2015

2000. Even though the HiperLAN standards share similarities with the IEEE 802.11 protocol family e.g. defining physical layer (PHY) and medium access control, the HiperLAN standards did not find a wide acceptance.

The IEEE released the first IEEE 802.11 protocol in 1997, which was subsequently ratified with the introduction of the IEEE 802.11a [11] and the IEEE 802.11b [12] protocol in 1999. However, the initial IEEE 802.11 protocol [13] was developed to support *best effort* traffic, such as Web or Email traffic, only, and therefore does not provide any quality of service (QoS). Due to an increasing demand of real-time multimedia traffic that require some level of QoS, the IEEE ratified a new IEEE 802.11e protocol [14] in 2005 which has subsequently been included in the new specification of the IEEE 802.11 protocol [4] in 2007. The IEEE 802.11e protocol extension enhances the IEEE 802.11 medium access control (MAC) by introducing two new access control mechanisms, the *enhanced distributed channel access* (EDCA) and the *HCF controlled channel access* (HCCA), both of which we will discuss in more detail in Chapter 2, and also allows the modification of several MAC parameters that were previously fixed. Consequently, it is now possible to differentiate between different traffic types or set different priorities to different stations (access point (AP) and/or wireless node), thus allowing the fine tuning of the system performance for specific environments.

With increased computing power and capabilities of mobile devices, and in particular smart-phones, it appears to be a natural progression to use voice over IP (VoIP) applications on wireless devices. Voice over IP already enjoys a widespread use in the consumer and the commercial space. For example, there are more than 180 VoIP providers in Australia alone [15], excluding popular VoIP applications such as Skype[4] and Google Talk[5]. One of the particular attractions of VoIP is the potential cost savings it can provide compared to traditional (copper-based) telephone services (PSTN[6]) [16] and cell phone communication.

Even though the use of VoIP over WLANs seems a natural development it has proven to be a challenging area, because the design of the IEEE 802.11 protocol [4] and in particular the *medium access control* (MAC) severely limit the number of VoIP calls that can be supported in a WLAN. We have shown that in an IEEE 802.11b WLAN the number of voice conversations is limited to 6 or 7 [17]; far fewer than the 100+ VoIP calls that would be expected based on simple bit rate calculations. Therefore the primary focus of this work is to study voice over IP traffic in wireless networks and provide solutions to overcome the limited VoIP capacity, as we will discuss in the next section in more detail.

1.1 Aim and Scope

In the previous section we have briefly outlined that the limited number of VoIP calls that can be supported in an IEEE 802.11 WLAN is a major drawback to the success of wireless voice over IP. As we will discuss in more detail later on, the limited number of VoIP calls is caused by the access point (AP) in an infrastructure WLAN, because all traffic to and from all wireless devices in the network has to pass through the central AP, which is a bottleneck. Hence the main aim of this work is to gain a better understanding of the interactions between VoIP traffic and the MAC protocol at a station that limits the number of VoIP calls. Based on our analysis we provide solutions that address the shortcoming and we show that these solutions are a viable option for improving both the VoIP capacity and the overall system performance.

Even though there have been numerous proposals in the literature to overcome the limited VoIP capacity, a variety of these proposals among other things require significant changes to hardware, software or the network protocol, or only provide a marginal increase in VoIP capacity. We discuss some of the proposals in Chapter 2 and highlight their individual advantages and shortcomings. We demonstrate that there are no proposals which significantly improve performance without requiring hardware and standards changes, which is the main

[4] http://www.skype.com
[5] http://www.google.com/talk
[6] Public switched telephone network

focus of this thesis.

In order to analyze and understand the reasons behind the limited VoIP capacity in an IEEE 802.11 infrastructure WLAN, in Chapter 3 we develop a detailed analytical model that captures the interactions between VoIP traffic and the QoS MAC protocol (EDCA) at a *station* (access point or wireless node). Using the insight we gain from the analytical model allows us to develop solutions to tackle the limited number of calls. In particular, the analytical model captures the behavior of the QoS mechanism of the IEEE 802.11 protocol, and hence allows us to fine tune the parameter settings such that an optimal solution is developed.

In our work we have made an explicit decision to approach this problem in such a way that any solution we propose can be implemented widely and rapidly. In particular, from our point of view, an ideal solution to address the limited VoIP capacity:

1. Does not require changes to hardware, software[7] or the wireless protocol and no additional hardware or specialized software is required,

2. Should allow a swift and uniform deployment with existing hardware, software and protocols,

3. Has to provide a significant increase in number of VoIP calls supported, rather than a minor adjustment,

4. Should increase the overall system performance rather than increase the performance of a single network device, i.e. the AP only,

5. Has to be versatile such that the solution can be used in different environments, i.e. different physical layers, and with different parameter settings, e.g. change to the QoS parameter of the IEEE 802.11 QoS mechanism, which we discuss in Chapter 2,

6. Should ensure a high level of VoIP call quality as measured using the *mean opinion score* (MOS) [18] or the $R-$value of the ITU-T E-model [19].

In this work we will focus on the VoIP capacity in terms of number of VoIP calls that can be maintained with an acceptable level of quality. Even though other measures such as UDP throughput [20, 21] may provide a more

[7]Software here means for example the VoIP client or software running on the WLAN infrastructure, e.g. the AP.

fine grained insight into the performance of wireless VoIP, using the number of VoIP calls is a more relevant measure of system performance. Nevertheless, we will also consider other performance metrics to support our results.

Our analytical model not only allows us to gain an insight into the interactions between VoIP traffic and the MAC protocol, such as the average number of transmission attempts required to successfully transmit a packet between two stations, it also allows us to study changes to the system behavior if *external*[8] parameters are changed. In Chapter 4 we study the buffering requirements of voice over IP traffic at a station. By analyzing the buffering requirements, we can gain a further insight into the VoIP-MAC interactions by inspecting the queue utilization, the delay and packet loss, and whether changes to the buffer size affects the system performance. We show that there is a minimum buffer size that maximizes the VoIP capacity in WLAN, and that VoIP capacity is independent of the buffer size, given this minimum buffer. To validate our finding that the number of voice conversations is independent of the buffer at a station, we modify the model developed in Chapter 3 to allow the investigation of the VoIP capacity when there is an infinite buffer space at a station. Based on the updated model we present a closed-form expression for VoIP capacity in an infrastructure WLAN. Using the insight that we can gain from the analytical model, we propose a novel VoIP capacity approximation formula that can be used to determine the VoIP capacity in conjunction with the adjustable QoS parameter of the IEEE 802.11e protocol [14].

Even though the current IEEE 802.11 protocol provides QoS to different voice streams and the IEEE continues to improve current and new standards, i.e. the IEEE 802.11n protocol [22], there are additional means to improve the VoIP capacity in an IEEE 802.11 infrastructure WLAN. In Chapter 5 we propose *dynamic* voice codecs in conjunction with traffic prioritization to gain a further VoIP capacity increase. We also focus on supporting the highest possible call quality for all voice calls. Based on the traffic prioritization and the dynamic voice codecs we propose the novel *dynamic codec with priority* (DCwP) scheme. To show the benefits of the proposed DCwP scheme we develop an analytical model based on the analytical model presented in Chapter 3. To evaluate the call quality, we use the ITU E-model.

In our early chapters we consider wireless networks where VoIP is the only traffic. However, in a real network the VoIP traffic has to compete for chan-

[8]External parameters are parameter settings that are not related to the channel access, such as PHY or changes at a station, e.g. queue management algorithm.

nel access with stations sending and receiving other traffic types. In Chapter 6 we investigate the impact of TCP streams on the VoIP capacity in WLAN. We begin by showing how the different channel access mechanism copes with multiple traffic types. Once we have established a base case, we study if a further VoIP capacity gain can be achieved using our proposed solution. However, we do not limit our analysis to the VoIP capacity only, and we also study how our proposed solution affects the TCP data flows, in terms of maximum aggregated throughput that can be maintained. Despite there being considerable work published on TCP flows over WLAN, we believe our analysis is the first to provide a complete picture that integrates uplink and downlink TCP flows, parameter changes to the TCP traffic and also changes to the priority of VoIP traffic.

1.2 Outline

The thesis is organized as follows. In Chapter 2 we provide background on Voice over IP in wireless networks. This chapter has two parts, i) a general overview of voice over IP and ii) details about the IEEE 802.11 protocol. In the first part we highlight the differences between server-based and peer-to-peer VoIP and discuss the different VoIP protocols such as the *Session Initiation Protocol* (SIP) [23] and the ITU-T H.323 [24] protocol. We then discuss different voice codecs and outline differences between *constant bit rate*, *variable bit rate* and *dynamic* voice codecs, before we provide details about the voice call quality and how it can be measured. In the second part we focus on background information regarding the IEEE 802.11 protocol. In particular we discuss the different network architectures before we focus on the *medium access control* (MAC) of the IEEE 802.11 protocol. This is followed by a summary on the Quality of Service (QoS) provisioning in the IEEE 802.11e protocol [14]. Finally we present the relevant literature in this area and identify the research this thesis addresses.

In Chapter 3 we discuss a solution to improve the limited VoIP capacity based on the IEEE 802.11 QoS mechanism. To show the benefits of our solutions, we develop a detailed analytical model. We validate our analytical results by simulation and testbed measurements, both of which are discussed in detail in Section 3.5. We first confirm the limited number of voice calls using our model, by comparing the results obtained with those in the literature. We then show the voice capacity increase that can be gained by our proposal and we show that our solution outperforms similar solutions described in the literature.

In particular, we identify that our solution is optimal in terms of voice capacity and overall system performance. We extend our analysis and show that the solution is versatile such that it can be used for variable bit rate voice flows. Before we provide a summary of this chapter, we investigate if a further capacity gain can be achieved when the proposed solution is used in conjunction with other QoS parameter introduced in the IEEE 802.11e protocol, and we show that our proposed solution is optimal and further changes to other QoS parameters will not result in a further capacity gain.

The impact of the buffer size is the main focus of Chapter 4. We show that the voice capacity is independent of the buffer size. Based on our findings we develop a closed-form expression for the number of voice calls that can be supported when there is no buffer limitation at a station and the proposed solution of Chapter 3 is used to improve the number of acceptable voice calls. We show that our model can be used to rapidly obtain the VoIP capacity even if variable MAC and/or QoS parameters are considered. Furthermore, our model also captures variable internal parameter changes such as changes to the collision probability based on the arrival rate that may not be possible with other alternatives [2,25] that use real testbed measurements. Also in this chapter we propose a novel VoIP capacity approximation equation based on the closed form expression. Using this approximation allows us to gain further insight into the VoIP capacity in WLAN. In particular it allows a further investigation into the upper bound of the voice capacity as discussed in Chapter 3.

In Chapter 5 we propose a novel scheme whereby we exploit a tradeoff between voice call quality and channel access priority to further improve the performance and quality of the voice calls in WLAN, compared to the solution used in Chapter 3. Whereas in our first proposal to improve VoIP capacity, all voice calls experience an unacceptable voice call quality once the WLAN becomes saturated, this novel scheme allows a further capacity gain while maintaining a high level of call quality. To show the benefits of this scheme, we derive an analytical model and use simulation. The detailed analytical model also allows us to study the interactions between the high and low priority VoIP traffic. In particular we provide details about the *internal* collision at a node, and show that the total collision probability can be obtained using a simple recursive expression. This is of particular interest, because the internal collision at a node has not been widely studied, and is commonly neglected [26].

Whereas the previous chapters were only concerned with VoIP traffic, in Chapter 6 we discuss the impact of concurrent TCP flows in the WLAN on the voice capacity. Here we will show the difference in voice capacity as well as TCP throughput for the different channel access functions as well as our proposed dynamic codec scheme of Chapter 5.

We conclude our work in Chapter 7 and provide an outlook on future research.

1.3 Contributions

Overall, our contributions in this work can be summarized as follows:

1. We develop a detailed analytical model to obtain the maximum number of voice calls an IEEE 802.11 infrastructure WLAN can support with an acceptable quality. A significant contribution of our model is the accurate modeling of the EDCA backoff mechanism. As there is a difference in the way the backoff counter is decremented in DCF and EDCA, compensation for the difference is required, which is captured by our model. Then using this model we confirm the limited voice capacity. We show that the channel access mechanism rather than bandwidth limits the voice capacity, by considering the higher bandwidth wireless networks based on the IEEE 802.11a/g standard. Here we show that even though the bandwidth in an IEEE 802.11a/g WLAN is approximately five times that of an IEEE 802.11b WLAN, the voice capacity does not increase by the same factor. Furthermore we show that our analytical model is versatile and can also be applied to investigate variable bit rate VoIP flows, and we show that assumptions made in [27] to obtain the voice capacity for variable bit rate voice flows induces imprecisions (Chapter 3).

2. Using our analytical model, we evaluate a solution based on the IEEE 802.11 QoS mechanism, and we show that a significant ($\approx 100\%$) voice capacity gain can be achieved with this solution. In particular, the solution provides priority to the AP only and is based on the transmission opportunity (TXOP), that is specified as the duration of uninterrupted channel access upon gaining access, and is defined by the $TXOPLimit$ parameter of the IEEE 802.11 MAC protocol. In particular we show that we can achieve the same capacity gain as reported in [2], but with a small value of $TXOPLimit$ parameter, that guarantees an overall higher system performance. Furthermore, we show that the voice capacity is still limited even if an arbitrarily large value of $TXOPLimit$ is set at the

AP, and that there exists an asymptotic value for the maximum number of voice calls in a WLAN. Specifically, we show that our proposed values of $T\epsilon XOP\epsilon Limit\epsilon$ are optimal in terms of VoIP capacity and overall system performance and that larger values of $T\epsilon XOP\epsilon Limit\epsilon$ will not result in increased capacity. We confirm that this optimal value maximizes the voice capacity in WLAN without compromising the performance of the wireless voice nodes. In particular, we show that setting larger than the optimal $T\epsilon XOP\epsilon Limit\epsilon$ parameter at the AP will trigger a bottleneck shift from the AP to the wireless voice nodes. Then we show that using an increased $T\epsilon XOP\epsilon Limit\epsilon$ with different sizes of the contention window has only a marginal impact on the voice capacity, and conclude that the $T\epsilon XOP\epsilon Limit\epsilon$ is an ideal parameter to use to increase the voice capacity in an IEEE 802.11 infrastructure WLAN (Chapter 3).

3. We investigate the impact of the buffer size at a station, and show that there is a minimum buffer size (K_{min}) with which the voice capacity in an IEEE 802.11 infrastructure WLAN is maximum and further increasing the buffer will not increase the number of calls that can be supported. In particular we show that the voice capacity is independent of the buffer size for buffer sizes $K\epsilon \geq / K_{min}$. Based on the $M/G/1/K\epsilon$ model developed in Chapter 3 we derive a queueing model with infinite buffer space ($M/G/1/\infty$), and show that both models are in good agreement for a wide range of parameters, supporting our claim that VoIP capacity in IEEE 802.11 WLANs is independent of the buffer size. Based on the $M/G/1/\infty/$ queueing model, we develop a closed-form expression for the number of voice conversation that can be supported. We show that a similar voice capacity gain can be achieved in this model, when access preference is given to the AP using the aforementioned $T\epsilon XOP\epsilon Limit\epsilon$ parameter to boost the VoIP capacity. Finally, we show that the buffer independence also holds if variable bit rate voice streams are considered. (Chapter 4).

4. Based on the aforementioned $M/G/1/\infty/$ queueing model, we propose a novel way to obtain the voice capacity in an IEEE 802.11 WLAN, and propose the *VoIP Capacity Approximation*. We show that this approximation matches results obtained using the $M/G/1/K\epsilon$ and the $M/G/1/\infty/$ model for a wide range of parameter settings, such as $T\epsilon XOP\epsilon Limit\epsilon$ or codec sampling rates. Additionally, the VoIP capacity approximation al-

lows us to gain further insight into the voice capacity in WLAN. We analytically confirm the asymptotic value for the voice capacity and show the interactions of VoIP packets in the uplink and downlink direction when the adjustable $TXOPLimit$ parameter of the IEEE 802.11 MAC protocol is applied at the AP (Chapter 4).

5. We propose a novel scheme based on dynamic voice codecs to reduce the overall contention in a highly congested WLAN. We show that this scheme improves the voice capacity in WLAN between 16% and almost 300% while maintaining an acceptable quality for all individual calls. Here we also use the ITU-T E-model to obtain the call quality for different network conditions. To show the benefits and performance of our proposed scheme, we develop a detailed analytical model to obtain the voice capacity in a multi-codec environment, that also takes the internal collision at the AP into account. For the latter, we show that the collision probability of a low priority packet can be obtained using a simple recursive expression (Chapter 5).

6. Finally we analyze the impact of TCP streams on the VoIP capacity in IEEE 802.11 infrastructure wireless networks. In particular, we first show the difference in VoIP capacity and TCP throughput when the channel access is controlled by DCF and EDCA. We then focus on improving VoIP capacity and TCP throughput. Here we show that using the $TXOPLimit$ solution to give preference to VoIP traffic not only increases the number of calls that can be supported, furthermore we show that this solution also protects in the TCP streams, leading to a higher aggregated TCP throughput. Finally, we evaluate the performance of our proposed dynamic codec with priority scheme when WLAN is shared between VoIP and TCP streams. We show that our proposed scheme is superior in terms of number of voice calls and aggregated TCP throughput when compared with the $TXOPLimit$ solution for different scenarios (Chapter 6).

2 Background

In this chapter we provide some required background information on voice over IP (VoIP) and the IEEE 802.11 protocol [4]. In particular, we discuss the different VoIP protocols, voice codecs and methods to evaluate the voice call quality. In the discussion of the IEEE 802.11 protocol we focus on the *medium access control* (MAC) protocol, because, as we will show later on, the MAC protocol has a significant impact on the performance of voice flows in WLANs. Additionally we outline some of the protocol improvements and in particular discuss the quality of service (QoS) provisioning in IEEE 802.11 WLAN. Furthermore we outline some of the challenges that need to be tackled when VoIP is used in wireless networks, and we provide a brief summary of some of the related work in this area.

This chapter is organized as follows. In Section 2.1 we provide a general overview of VoIP that includes the different VoIP network topologies, protocols and voice codecs. We discuss how VoIP quality can be measured using the common ITU E-model and the mean opinion score. This is followed by an overview of IEEE 802.11 WLANs in Section 2.2, that includes the different architectures and the wireless medium access control, followed by some details about provisioning of quality of service in WLANs. In Section 2.3 we present a literature review focusing on the limited VoIP capacity, proposed solutions to improve the number of VoIP calls, as well as literature discussing buffer requirements, dynamic voice codecs and the impact TCP data flows have on the number of VoIP calls that can be supported. We conclude with a brief summary.

2.1 Voice over IP - An Overview

Voice over IP (VoIP) or *internet telephony* is one of the fastest growing services on the Internet today [28–30]. The market share of long distance calls of Skype for example has risen from 8% in 2008 to above 12% in 2010 [31]. Furthermore, the number of Skype calls is increasing, whereas the number of phone calls using the common *public switched telephone network* (PSTN) is stagnating [32, 33]. Even though security of VoIP services, and in particular Skype is still a concern [34–37], many consumers and businesses now use VoIP services to lower their communication costs. For businesses, the cost reduction can be as significant as 50% [16]. In the consumer space real cost benefits are provided by so-called VoIP flat-rates [38] where a user pays a monthly fee and is not charged for any individual calls. Examples of such flat-rates are the Vonage[1] "U.S. Canada Unlimited" plan or PennyTel's[2] "Crazy Talk" plan. Even though there is an increasing number of VoIP providers, for example, there are more than 300 VoIP providers in Australia [15] alone, Skype appears to be predominant in the VoIP market. In a recent VoIP market analysis [39] it is reported that 50% of the participants in that study use Skype, exceeding the market share of other popular providers such as Verizon or AT&T in the United States.

2.1.1 VoIP architectures

Whereas a PSTN is based on an hierarchical topology as shown in Fig. 2.1 and the calls are routed through the hierarchy from the caller to the callee, voice over IP uses the underlying IP network, i.e. the Internet, to route the call, and thus can be independent of the hardware hierarchy. However, depending on the VoIP implementation there are two types of VoIP architectures - a *client-server* architecture and a *peer-to-peer* (P2P) architecture, both of which we will now briefly discuss.

[1] http://www.vonage.com
[2] http://www.pennytel.com.au

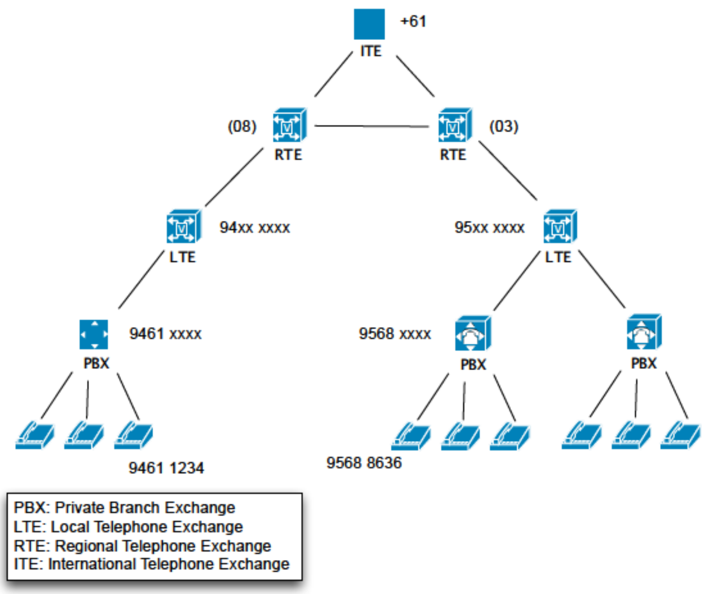

Figure 2.1: Public Switched Telephone Network (PSTN)

Client-server VoIP

A client-server VoIP architecture shares some similarities to a PSTN. In this architecture the individual VoIP clients (hardware and/or software) are connected to a corresponding VoIP server, similar to a PBX. The VoIP server is then connected to a different network such as the Internet or a private LAN. Such an architecture is common where a VoIP infrastructure replaces the PSTN, such as in office environments. Examples of such a topology is an infrastructure based on the Cisco Call Manager[3] or the free iptel.org [4] service based on the *SIP Express Router* (SER)[5], where the different VoIP clients connect to the server for user authentication as well as additional services, such as call restrictions.

In a client-server VoIP architecture the voice calls are commonly *end-to-end* (E2E), meaning voice stream is directly exchanged between the VoIP clients. As we will discuss later on, this is not necessarily the case in peer-to-peer VoIP. As depicted in Fig. 2.2, the VoIP server can also share a connection with PSTN gateways to allow a seamless integration of VoIP with the PSTN.

[3] Cisco Unified Communications Manager - http://bit.ly/n9ix7R
[4] http://www.iptel.org
[5] http://www.iptel.org/ser/

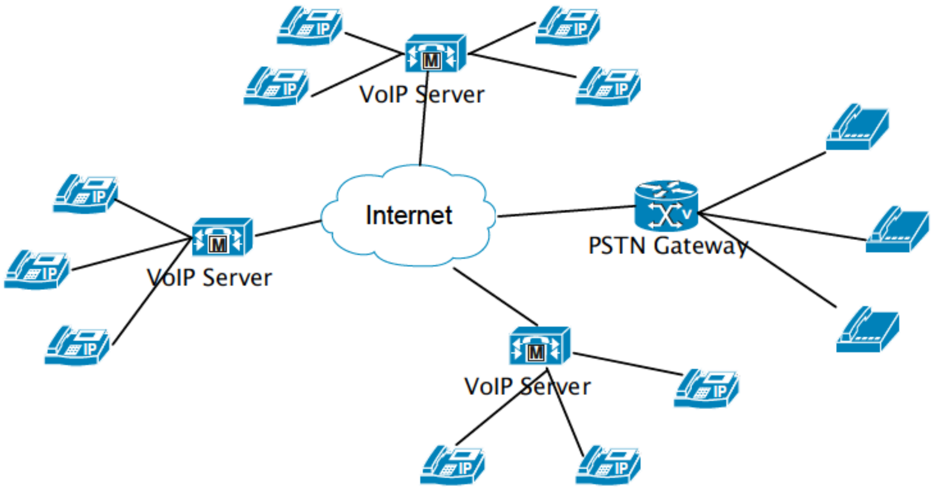

Figure 2.2: A client-server VoIP architecture

Peer-to-peer VoIP

A peer-to-peer (P2P) VoIP topology does not require a central VoIP server, since the connection between the different nodes is established using an overlay network [40–42]. The most popular P2P VoIP application is Skype[6]. The advantage of P2P VoIP is that the communication between the nodes relies solely on the underlying Internet infrastructure, and hence there is no need for dedicated VoIP infrastructure such as VoIP servers or gateways. Because there is no dedicated VoIP infrastructure, services such as Skype can offer free calls between their users.

Whereas early peer-to-peer applications like Napster[7] used a centralized directory server to discover the resources, modern P2P applications, including Skype, use a decentralized directory infrastructure. In Fig. 2.3[8] we show a stylized decentralized directory Skype overlay network. As shown, a variety of nodes connect to a group-leader [40] (Skype supernode), who in turn are interconnected to route traffic between them. The impact of supernodes on the network has been studied in [43–45].

2.1.2 Protocols

Irrespective of the VoIP architecture, the basis of Voice over IP calls is a set of communication, media and transport protocols. In Figure 2.4 we show a

[6] https://support.skype.com/en/faq/FA10983/What-are-P2P-communications

[7] http://www.napster.com

[8] Note that the "S" is the common Skype logo and represents an individual Skype node (small) and a super-node (large)

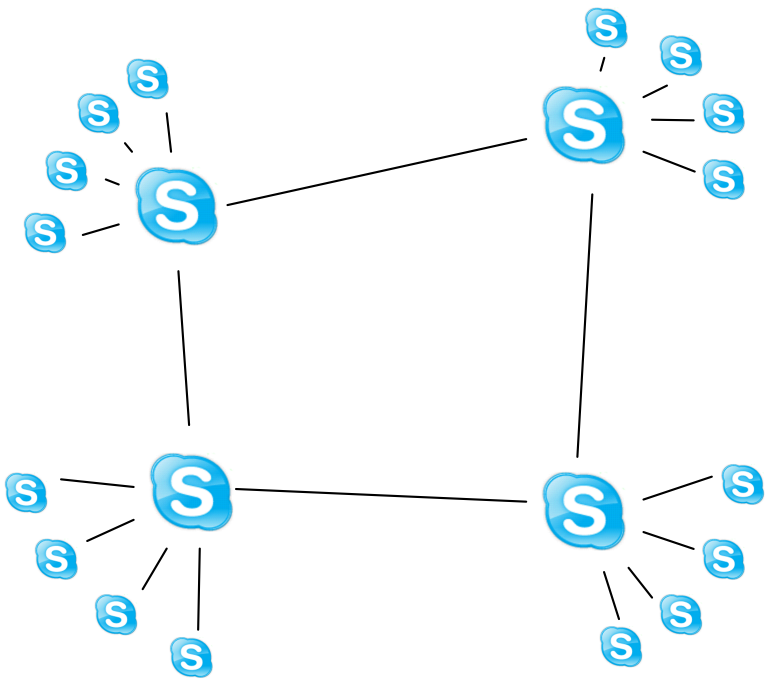

Figure 2.3: Skype peer-to-peer infrastructure

stylized diagram of how VoIP works (once the initial connection has been established). In this scenario we assume that Bob is talking to Alice. Bob first initiates the call using a communication protocol (not shown). The spoken phrase "What is VoIP" is digitized by the voice codec (media protocol) and then transmitted as IP packets (transport protocol). Once Alice receives the IP packets, the voice codec decodes the information and replays the message. If either Bob or Alice hang-up, the communication protocol will terminate all open connections between Bob and Alice.

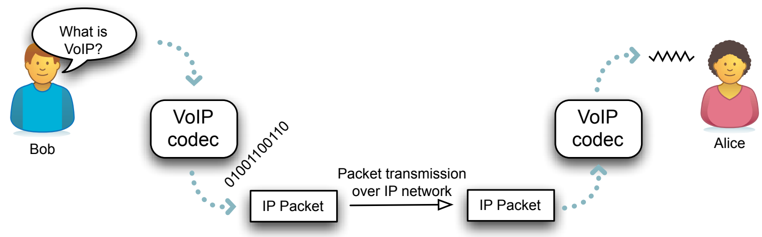

Figure 2.4: Stylized diagram of Voice over IP

Each protocol plays a vital part in the voice over IP communication, and we will discuss each protocol type in the following.

Communication protocol

The communication or *signaling* protocol of voice over IP is used to manage the call. This includes call creation, call modification and call termination. Depending on the protocol it may also manage the VoIP service registration and authentication.

There are a variety of different VoIP communication protocols. However, the two predominant protocols today are the *Session Initiation Protocol* (SIP) [23] and the ITU-T H.323 protocol [24]. Both protocols have been widely implemented by a variety of manufacturers and are used in a client-server VoIP architecture in conjunction with SIP proxies and SIP registrars or H.323 gatekeepers as well as gateways to connect to the PSTN. Even though the additional VoIP infrastructure (SIP proxy/H.323 gatekeeper) are not mandatory, they are commonly used for user authentication and allowing access to the VoIP network and provide additional VoIP services such as ENUM[9] [46, 47] or media proxy services required for firewall and NAT[10] traversal [48].

Because of the popularity of peer-to-peer networks different communication protocols have been developed. Whereas VoIP applications such as Skype use a proprietary communication protocol where only a few details are known [49–51], other applications such as Google Talk use an open XMPP/Jabber protocol [52] as a peer-to-peer communication protocol.

In spite of the different VoIP architectures, the communication protocol in client-server and peer-to-peer VoIP work in a similar way and the process of a VoIP call establishment is stylized in Fig. 2.5. As shown, a call is initiated by a call request, and answered by a call answer, and the call is established (media exchange), before a call termination request terminates the voice call. Note that some VoIP applications allow the characteristics of the call, such as the voice codec to be modified during the call.

Media protocol

The media protocol is implemented by way of a so-called *voice codec*. A voice codec converts an analog audio signal to a digital audio signal and vice versa. There are many different voice codecs available, each designed to suit specific requirements or applications. For example, the AMR (Adaptive Multirate) voice codec[11] is the mandatory voice codec for 2.5G/3G wireless communica-

[9] E.164 NUmber Mapping
[10] Network Address Translation
[11] http://www.3gpp.org/ftp/Specs/html-info/26090.htm

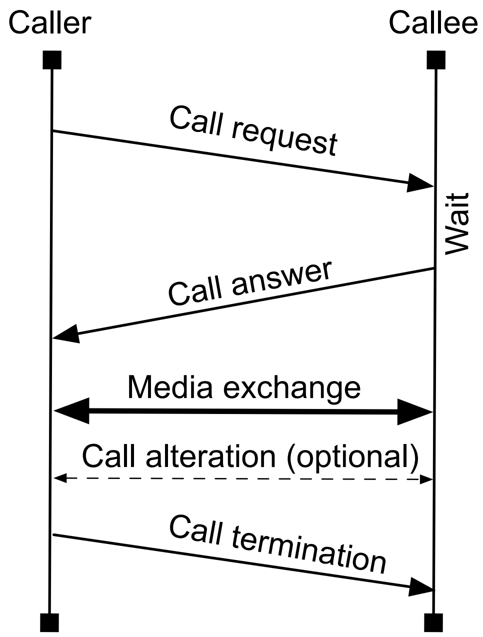

Figure 2.5: VoIP call control

tion based on a GSM[12] network and has been standardized by ETSI. Initially, GSM networks used a so-called *Full Rate* voice codec [53] based on the RPE-LTP (Regular Pulse Excitation Long-Term Prediction) speech coding. However, newer GSM systems use different codecs such as EFR (Enhanced Full Rate), ACELP (Algebraic Code Excited Linear Prediction) or CELP-VSELP (Code Excited Linear Prediction - Vector Sum Excited Linear Prediction). Other codecs such as the ITU G.711 [54] or ITU G.729 [55] voice codec were designed for traditional telephone systems to guarantee a high call quality. These codecs have also been adopted in VoIP telephony. With the increased use of VoIP, other high quality codecs such as G.722.1 [56] have been developed to support additional features. The G.722.1 codec for example, provides 7Khz stereo quality for VoIP. We discuss some of the common voice codecs and some of their properties in more detail in Section 2.1.3.

Transport protocol

Finally, the transport protocol is used to transmit the voice data over the IP network. As voice over IP data is time sensitive, UDP/IP is commonly used as the transport protocol. This is because UDP is better suited to voice transport

[12] GSM stands for Global System for Mobile Communication, and is a standard for cellular communication

than TCP since it has a lower latency. To provide some reliability, some VoIP applications use the *real-time transport protocol* (RTP) [57], which adds timing information as sequence numbers. This allows the receiver to determine delay adequately and identify out of order packets.

2.1.3 Voice codecs

In the previous section we provided a brief overview of the media protocol and outlined that this protocol is implemented by way of a voice codec. In this section we provide more details on common voice codecs. To differentiate between the different voice codecs and their individual properties, we classify each voice codec either as *static*, *variable* or *dynamic* voice codec, which we define in the following.

Static voice codecs

A static voice codec is a voice codec that generates a *constant bit rate* (CBR) voice stream. A static voice codec generates a voice flow with a constant bit rate, where the voice packets have an equal payload size and are sampled at a constant interval.

An example of a constant bit rate voice codec is the ITU G.711 voice codec [54], also known as *Pulse Code Modulation* (PCM). This voice codec has a sampling rate of 8000 samples per second and each sample is encoded using 8 bits. Therefore the G.711 voice codec has a bit rate of $8000 \times 8 = 64000 = 64$kbit/s. Depending on the sampling interval of the codec, the ITU G.711 voice codec generates 100 packets/s, each with a payload size of 80 bytes when a 10 ms sampling interval is used, or 50 packets/s with a payload size of 160 bytes for a 20 ms sampling interval. Different VoIP codecs generate a different constant bit rate voice stream as it depends on the payload size and the sampling rate, and is a tradeoff between efficiency and latency. In particular, the bit rate of a codec can be calculated based on the payload size L and the sampling interval Λ using $R_b = (L \times 8)/\Lambda$. In Table 2.1 we show the different bit rates for different voice codecs.

Variable voice codec

Unlike a CBR voice codec, a variable voice codec generates a *variable bit rate* (VBR) voice stream. These codecs generate a voice packet of a fixed payload size only if there is actual data to be sent, i.e. a user is not silent. These

Voice codec	Bit rate
G.711	64 kbit/s
G.729	8 kbit/s
G.723.1	5.3/6.3 kbit/s
G.726	24/32 kbit/s
G.722	64 kbit/s
iLBC	13.33/15.2 kbit/s

Table 2.1: Bit rate of different voice codecs

codecs support the *discontinued transmission* (DTX) of the *voice activity detection* (VAD) to recognize if a user is talking or not (generating data or not). Note that the G.729 Appendix B and the G.711 Appendix II define DTX and VAD for G.729 and G.711 voice codecs, thus supporting VBR voice streams for both types of codecs.

During a regular voice conversation between two parties A and B, there are times when A is talking (talk-spurt) and B is listening, when A is listening while B talks (talk-spurt), when A and B are silent (mutual silence) and there are times when A and be B will talk at the same time (double-talk). Due to this "on-off" behavior the codec generates a variable number of packets, which results in a variable bit rate voice stream. The ITU-T P.59 [3] standard provides a guideline for the duration of each of the scenarios above. In our analysis in later chapters we provide more details and apply the recommended values of the ITU-T P.59 protocol when variable bit rate voice streams are considered.

Dynamic voice codecs

Research and development has lead to new types of codecs, which we define as *dynamic* voice codecs. As discussed in the previous section, a VBR codec generates a variable bit rate voice stream by transmitting packets only when there is data to send. However, the payload size of the packets in a VBR stream does not change. For example, each packet has an 80 byte payload if a G.711 voice codec[13] with a 10 ms sampling rate is used with DTX. A dynamic codec on the other hand not only generates a voice stream with a variable number of packets, but the payload size can also be adjusted to accommodate changes in network conditions, such as packet loss or a high level of contention in WLANs. These new codecs are designed to provide a high level of quality in all types of networks, from dial-up to fiber connections. An example of a dynamic voice

[13]The G.711 Appendix II defines the DTX algorithm which used the *voice activity detection* (VAD) and *comfort noise generation* (CNG) to reduce the bandwidth use.

codec is SILK [58] used in Skype version 4. The SILK codec has been designed such that it can adjust its bit rate, packet rate, the packet loss resilience and uses DTX to cope with variable changes in the network condition.

Summary

In summary, each voice codec generates a different voice stream with different properties. We show an example of the difference in packet flow of static, variable and dynamic voice codecs in Fig. 2.6. Note that the numbers indicate the payload size in bytes, i.e. 80 bytes. As shown, the inter-arrival times of the static packets is constant, whereas the inter-arrival time for the variable and dynamic codecs change. Also note that the payload size for static and variable

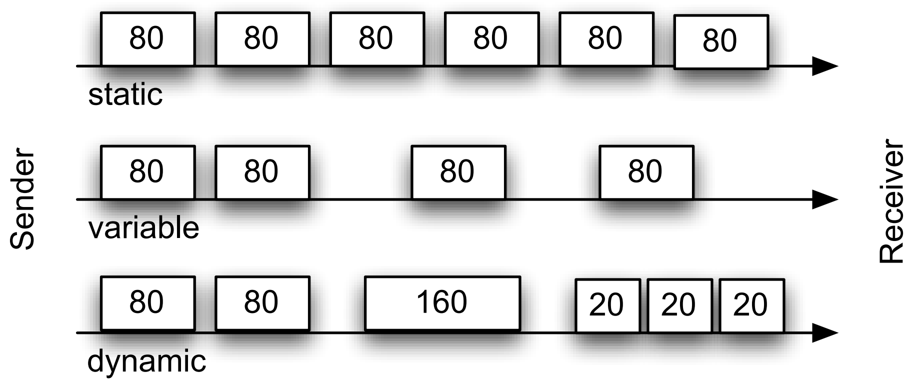

Figure 2.6: Stylized packet flow of a static (top), a variable (middle) and a dynamic (bottom) voice codec.

2.1.4 Voice call quality

In the previous sections we have discussed some of the requirements and background information on voice over IP. However, a successful VoIP call also depends on the user perceived quality of the call, such as the clarity of a voice call or the lag of the received voice. As in a VoIP conversation the spoken voice is digitized and transmitted over an IP network, the quality of the received voice may be reduced, depending on delay or packet loss. Beside packet loss, delay or jitter, there are also other interferences that reduce the user perceived call quality. For example, “echo” in a connection, whereby a speaker can hear its own words as an echo can be perceived as annoying and hence the user perceived call quality may be unacceptable. Also, varying volumes may also be interpreted as bad audio quality.

In Fig. 2.7 we show the six different levels of user satisfaction ranging from “Very satisfied” to “Not recommended” based on the *mean opinion score* (MOS) and the $R\epsilon-\not{v}$alue of the ITU E-model, both of which we discuss in m

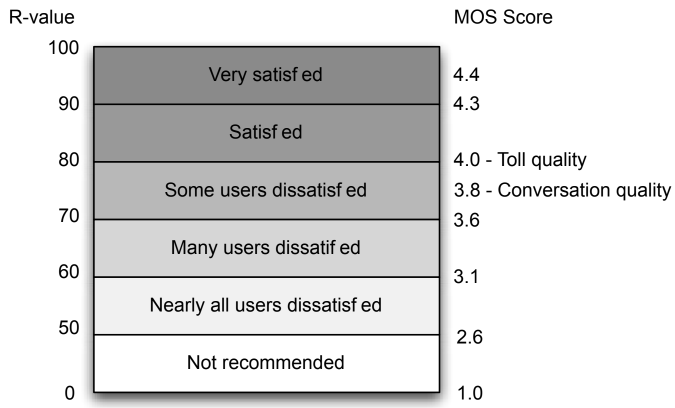

Figure 2.7: User satisfaction comparison measure as MOS and R-value

To evaluate the performance, efficiency and quality of voice over IP, different measures can be used to determine a level of voice call quality. These measures should incorporate the different quality impairments as we discuss later on as well as the user perception. This is because each user has a different level of tolerance of what is an acceptable level of delay for example.

In the following we discuss the different quality impairments and how they affect the user perceived call quality. This is followed by an overview of two common means of obtaining a quality measure of voice calls, namely the E-model defined in the ITU G.107 recommendation [19] and the mean opinion score (MOS) [18].

Quality impairments

As previously outlined, the user perceived voice call quality can be affected by different impairments. In particular we will focus on

- Packet loss
- Delay
- Jitter

- /Codec impairments

Packet loss

The impact of packet loss on voice quality has been widely studied for example in [59–63] and references therein. Packet loss in the network can have a significant impact on the voice quality and depends on whether a single packet loss event occurs, or whether a burst of packets is lost. Irrespective of the type of packet loss, packet loss results in the voice data being incomplete which can lead to a "choppy"[14] audio (voice) playback. As voice traffic is commonly transported using the UDP/IP transport protocol, there is no retransmission of lost voice packets, thus resulting in call quality deterioration, depending on the severity of the packet loss. Whereas low packet loss (e.g. $<\epsilon 2\%$) may not be audible to the receiver, higher level of packet loss (e.g. $>\epsilon 2\%$) may be audible, up to a point where the call quality is no longer acceptable [17, 64–66].

To compensate for packet loss, some voice codecs use *forward error correction* (FEC) [67] or *packet loss concealment* (PLC) [68–71]. These codecs may provide an acceptable call quality even if high ($\gg /2\%$) packet loss occurs [72].

Delay

During the transmission of packets from the sender to the receiver, packets experience different delays, for example delays at a router, the backoff delay caused by the MAC mechanism or queueing delays at intermediate nodes, such as at an AP in an infrastructure WLANs. The ITU-T G.114 standard [19] specifies that delays of up to 400ms can still provide an acceptable call quality. In particular, it is defined that the call quality is "excellent" for delay of up to 150 ms, still "acceptable" for delays between 150 ms and 400 ms and whenever the delay exceeds the 400 ms threshold, the call quality is no longer acceptable ("poor"). Different acceptable delays are used in the literature. For example, a 60 ms delay bound is applied in [27], whereas [73] and [64] use a 150 ms and 200 ms delay budget. The difference in delay budget is because the G.114 standard defines the delay as the *mouth-to-ear* delay, which includes the entire path between sender and receiver, whereas in the literature the delay bound is often set for the one-hop WLAN, e.g. for transmission between the AP and the wireless node and vice versa. Also note that the acceptable delay depends on the users' tolerance of what is a tolerable delay.

[14] Example audio files with different levels of packet loss and other impairments can be found at http://www.voiptroubleshooter.com/sound_files/

Jitter

Jitter, or the *packet delay variation* (PDV) is another network impairment of the call quality. In ideal conditions, a CBR VoIP codec generates a stream of data packets that are transmitted at regular intervals and therefore have a constant delay and inter-arrival time, that is a PDV of zero. Due to changing network conditions such as an increasing level of contention in the WLAN, the packet inter-arrival times change. To compensate for these variations, VoIP applications use a play-out buffer to smooth any larger changes in the inter-arrival time. It was shown that modern VoIP applications such as Skype or Google Talk can adapt the play-out buffer size to some degree, to compensate for larger fluctuation in inter-arrival time, traffic bursts and out-of-order packets [74]. Packets arriving past their play-out time are discarded on the node resulting in *local* packet loss. Note that VBR and dynamic voice flows have a higher jitter due to the flow behavior.

Even though jitter is an important factor for the voice call quality, our analysis using simulation and testbed measurements have shown that commonly the jitter is acceptable such that any variable changes in the inter-arrival of VoIP packets can be smoothed by the play-out buffer at the receiver. In particular, the observed jitter during our experimental analysis was well below the play-out buffer size for Skype, Google Talk and MSN Messenger as reported in [74]. Note however that the total delay may be increased due to the buffering of packets in the play-out buffer at the VoIP client and is dependent on the play-out buffer size [75].

Codec impairments

Beside packet loss, delay and jitter, which we refer to as *network impairments*, the call quality can also be reduced due to *codec impairments*. For example, [76] lists seven different sources of delay. However, only the coder (processing) delay, the algorithmic delay and the packetization delay are part of what we consider a codec impairment, whereas the serialization delay, the queueing/buffering delay, the network switching delay and the de-jitter delay are related to the transmission of the VoIP packet as outlined in the above sections. Nevertheless, these codec impairments impact on the VoIP quality and depend on the underlying voice codec and equipment. For example, the best case coder delay, i.e. the time required for the digital signal processor to compress a block of PCM samples, for a G.729 voice codec is 2.5 ms, whereas the worst case is 10 ms [76]. Similarly for the G.723 voice codec the best case is 5 ms and 20 ms for the worst case, respectively. Then depending on the scenario, the different codec impairments, e.g. the delays, can be a significant overhead, reducing the VoIP quality by increasing the overall delay.

Measuring the voice call quality

In the previous section we highlighted how network and codec impairments can affect the user perceived voice call quality. The overall call quality can be understood as a function of the *listening quality*, the *conversational quality* and the *transmission quality* [77]. To quantify the voice call quality, different methods can be used. Two of the most common quality measures are the R-value defined in the ITU E-model [19] and the mean opinion score (MOS) defined in the ITU P.800 standard [18]. Even though there are other call quality evaluation techniques, such as the *perceptual evaluation of speech quality* (PESQ) [78], we will focus on the R-value and the MOS, as they are predominantly used in the literature.

ITU-T E-model

The ITU-T E-model [19] specifies a measure of the user perceived call quality based on network metrics such as delay, loss as well as codec impairments. The measure of quality is expressed as a so-called R-value ranging from 0 for the lowest quality to 100 for the highest quality. The R-value can be calculated using [19]

$$R = R_0 - I_s - I_d - I_e + A$$

where R_0 is the basic signal to noise ratio (SNR) [19, Section 3.2], I_s specifies quality impairment due to non-optimum loudness and signal-correlated noise, I_d is an impairment caused by network delay, i.e. echo, I_e is the codec impairment and A defines an *advantage factor*, which can be understood as a tolerance or expectation level of a user for the call quality. An example for the A value, is that a user has a different expectation of call quality if a cell phone or a fixed-line service is used. Note that details about each individual term of the above equation is defined in [19], and that some default values for different voice codecs are given in [19, 79]. In Table 2.2 we provide an overview of the different R-values and the corresponding call quality levels.

An advantage of the ITU E-model compared to MOS is that it takes network measurements such as delay and packet loss explicitly into account, which is not

R-value	Quality level
90	Excellent
80	Good
70	Fair
60	Poor
50	Bad

Table 2.2: Voice quality level and R-values of the ITU E-model

the case for MOS. This allows a quality measure to be obtained during an active VoIP conversation. For example, the "technical call information" in Skype provides details about packet loss, delay and jitter, which could be used to derive the quality measure R. As information about the assumed user perceived call quality can be directly derived, the VoIP service provider can immediately make changes to the network if the indicated call quality is considered no longer acceptable, if for example if the delay exceeds a predefined quality threshold.

In Fig. 2.8 we show the $R\epsilon-\not{v}$alue for a G.711 voice codec with increasing network delay and different values of packet loss. As shown, with an increasing network delay the $R\epsilon-\not{v}$alue decreases, until user perceived call quality becomes unacceptable at some stage.

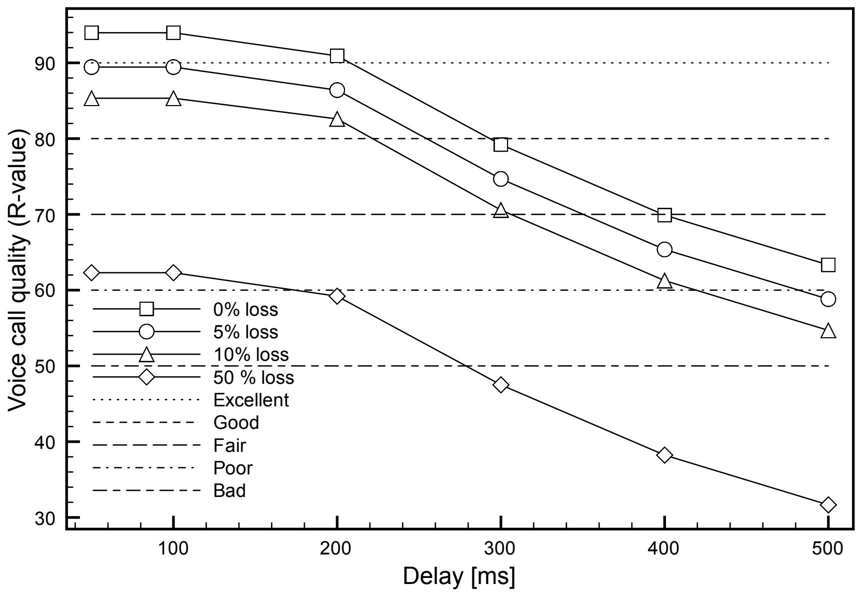

Figure 2.8: $R-\not{v}$alue of a G.711 voice codec with increasing delay and different level of packet loss

Mean Opinion Score

Throughout the literature, the *mean opinion score* (MOS) defined in the ITU

P.800 standard [18] in one of the most commonly used quality measures. The MOS is the result of an *absolute category rating* (ACR) test, whereby a number of listeners each rate the quality of a replayed audio/voice stream based on a five-rate scale from 1 (lowest quality) to 5 (highest quality). The individual responses are then averaged to obtain the final MOS for a particular voice/audio stream. In Fig. 2.9 we show an example of an ACR test for 250 listeners. As shown, in this example the majority of listeners rated the quality as “good” (MOS = 4), and the second highest number of listeners rated the audio/voice quality only as “fair” (MOS = 3). Based on this example, an overall MOS of 3.5 can be derived and the quality can be consider “fair” to “good” as shown in Table 2.3. Note that the P.800 standard [18] also defines the environment in which the quality measurement takes place. For example, it is specified that the recording of the audio/voice sample should be conducted in an environment where the background noise is below 30 dB, and that the recording device such as a telephone is calibrated based on the ITU P.64 recommendation [80]. Additionally, the audio source should consist of simple short sentences, such as the Harvard sentences [81] that are available online in the Open Speech Repository[15] to allow consistency between different tests and environments.

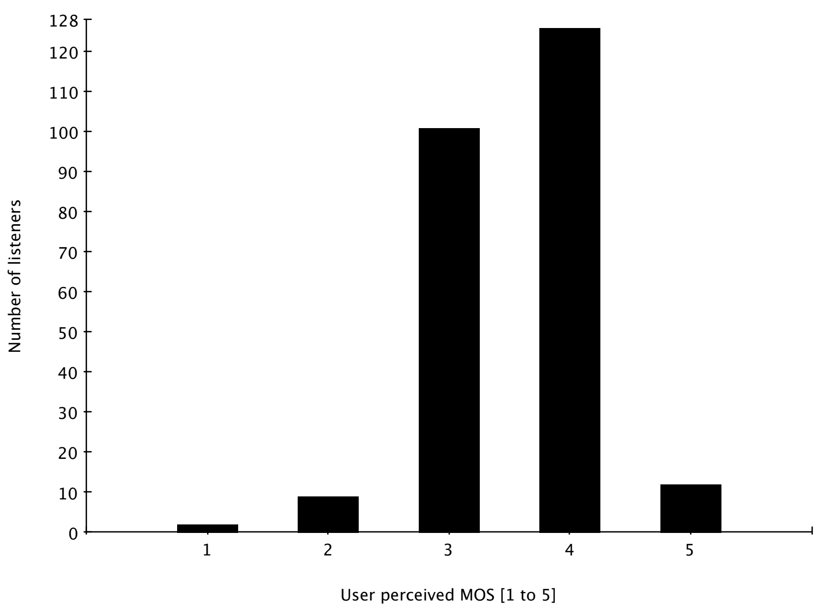

Figure 2.9: Example of ACR results for MOS

However, as the mean opinion score is a subjective measure of voice quality it can not provide a real time analysis of the voice quality. In particular, the

[15] http://www.voiptroubleshooter.com/open_speech/

MOS does not explicitly includes packet loss or delay, as it is done in the ITU E-model we discussed previously.

Nevertheless, the MOS is the most common indicator for the voice quality. Therefore, as part of the ITU E-model, an anticipated MOS score can be derived based on the $R-$value obtained using the E-model. In particular, the corresponding MOS based on the $R-$value can be calculated using [19]

$$\text{MOS} = 1 + 0.035R + R(R-60)(100-R)7e^{-6}.$$

In Fig. 2.10 we show the $R-$value and its equivalent MOS obtained using the above equation.

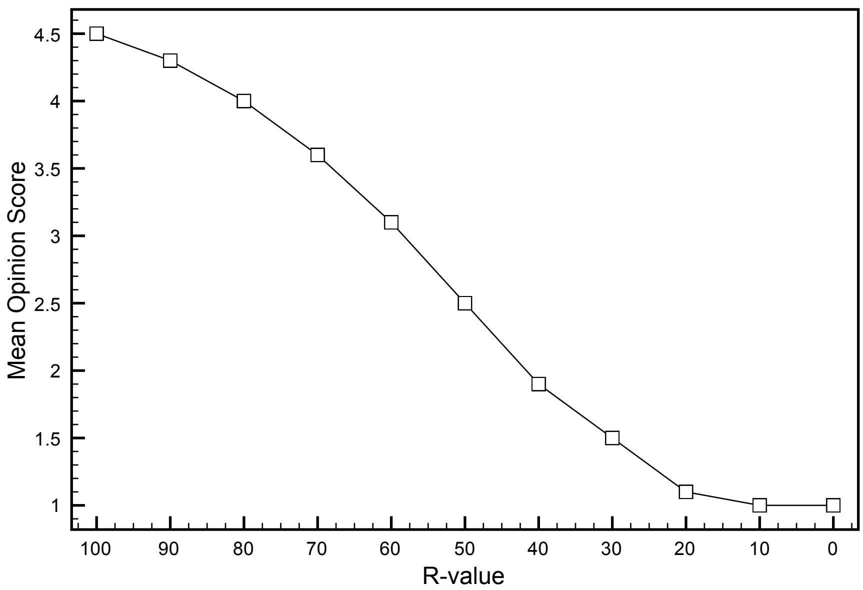

Figure 2.10: R-value and MOS comparison

Based on the G.114 recommendations of packet loss and delay, and using the above equation to obtain the MOS using the $R-$value, we show the MOS of a G.711 voice stream with a 10 ms sampling rate for increasing packet loss and different ranges of delay d in Fig. 2.11. Observe that whenever the packet loss exceeds 2%, the call quality will drop to a lower quality level, e.g. from 4.4 to 3.9 ("good" to "fair") for $d < 150ms$ or from 3.5 to 2.7 ("fair" to "poor") for $d > 400ms$. Similarly, in Fig. 2.12 we show the impact of packet loss on the MOS. As shown, the MOS of a G.711 voice stream with increasing network delay is still acceptable (MOS > 3) for delays of up to 400 ms. However, for delays $d > 400ms$ the MOS will drop to below 3 ("poor").

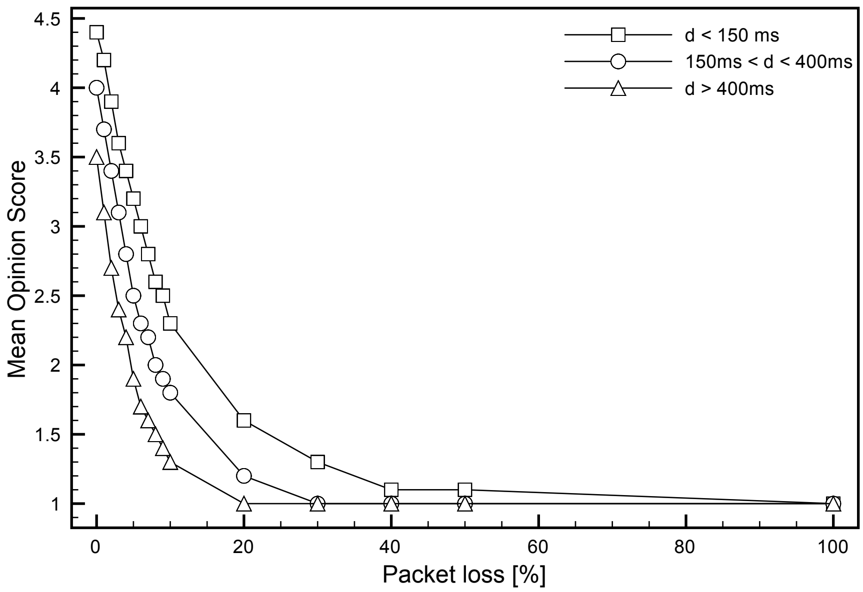

Figure 2.11: MOS of a G.711 voice stream with increasing packet loss and different level of network delay

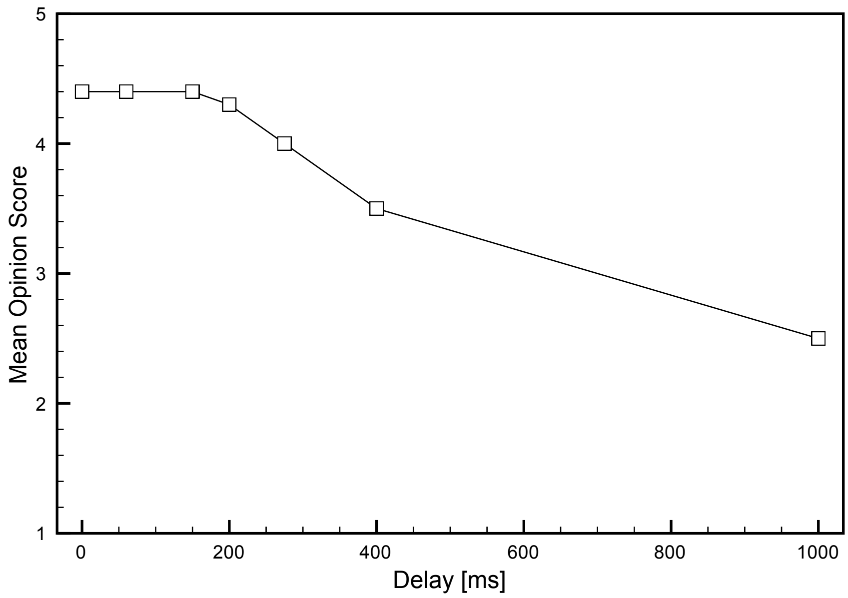

Figure 2.12: MOS of a G.711 voice stream with increasing delay and 0% packet loss

MOS	Quality level
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Table 2.3: MOS and corresponding user perceived call quality

2.2 IEEE 802.11 Wireless LAN

In this section we provide an overview of some important aspects of an IEEE 802.11 WLAN. We will briefly discuss the two types of wireless architectures - an ad-hoc and an infrastructure WLAN, and outline the requirements to transmit data packets using the wireless channel. For the latter, we will focus on the *medium access control* (MAC) of the IEEE 802.11 protocol, because, as we will show later on, the IEEE 802.11 MAC plays a crucial part in the performance of VoIP in WLANs. In particular, we will discuss the different channel access functions and highlight some enhancements made by the IEEE groups to the IEEE 802.11 protocol to provide *quality of service* (QoS) in WLANs.

2.2.1 WLAN architectures

An IEEE 802.11 WLAN can consist of two types of networks, an ad-hoc WLAN and an infrastructure WLAN. In this section we provide a brief overview of the two types.

IEEE 802.11 ad-hoc WLANs

In an *ad-hoc* or *peer-to-peer* WLAN wireless nodes can communicate directly with each other forming a wireless network consisting only of wireless nodes without the need of a central access point. These networks are useful in situations where the central access point is not available or not required, for example, when sharing a file between participants during a meeting. Ad-hoc WLANs are not widely used for PC-to-PC communication, however, in purpose build networks such as sensor networks or VANETs (vehicle ad-hoc networks), ad-hoc WLANs are common. In particular, with an increased demand of wireless mesh networks [82], we anticipate that more ad-hoc WLANs will be deployed. In Fig. 2.13 we show an example of a simple five node ad-hoc WLAN, where the dotted lines indicate that the nodes are within direct communication range.

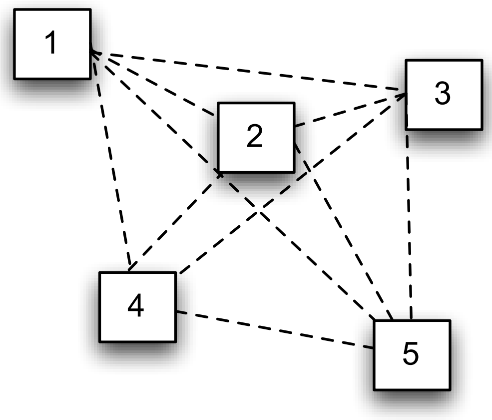

Figure 2.13: Stylized ad-hoc WLAN

IEEE 802.11 infrastructure WLAN

The second type of an IEEE 802.11 WLAN is an *infrastructure* WLAN. In this type of network, the wireless nodes require a central access point (AP). All communication in an infrastructure WLAN involves the AP, irrespective whether a wireless node communicates with a node outside or within the WLAN. Infrastructure WLANs have been widely deployed in home and business networks. Their widespread use is because the AP acts as a bridge between the wireless and the wired domain, allowing Internet access to wireless users. In Fig. 2.14 we show an example of a simple infrastructure WLAN with wireless and wired nodes. As shown, whenever wireless nodes communicate with each other (red lines), the communication path involves the AP. Similarly, when the wireless nodes communicate with a node outside the WLAN (blue lines). We will show later that the use of a central AP is a fundamental cause of some severe limitations in the performance of VoIP in IEEE 802.11 WLANs.

2.2.2 Wireless Medium Access Control

In order to transmit data in an IEEE 802.11 WLAN, a *station* (wireless node and/or AP) has to access the wireless channel. Only one station can access the channel at any given time. To control access to the channel, the IEEE 802.11 WLAN uses a *medium access control* (MAC) protocol to control the channel access, that is the focus of this section.

The IEEE 802.11 protocol [4] defines a set of *channel access functions*. In particular it defines the *distributed coordination function* (DCF) for the con-

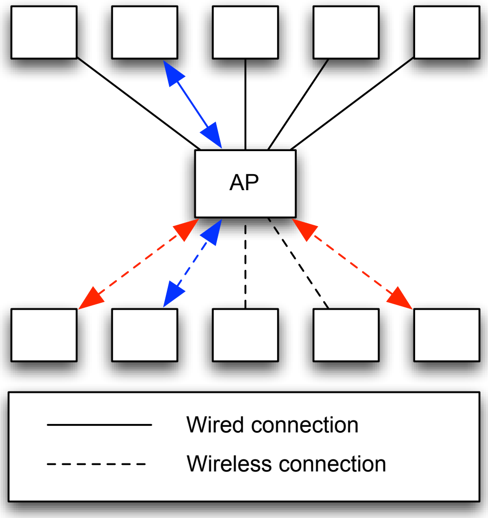

Figure 2.14: Stylized infrastructure WLAN

tention based channel access and the *point coordination function* (PCF) for the contention free channel access, both of which are designed without quality of service (QoS) provisioning. In 2005, the IEEE ratified the IEEE 802.11e protocol [14] that extends the medium access control to support QoS. It further extends the MAC by two additional channel functions, the *enhanced distributed channel access* (EDCA) and the *hybrid coordination function controlled channel access* (HCCA).

The Distributed Coordination Function (DCF)

The distributed coordination function (DCF) is the contention based channel access in an IEEE 802.11 WLAN, based on the CDMA/CA (Carrier sense multiple access with collision avoidance) mechanism [83, 84]. To transmit a packet in a DCF controlled wireless network, a station first has to determine the state of the wireless channel. If the wireless channel is sensed busy, due to active transmissions, the station defers its transmission, until the channel is sensed idle for the duration of the *distributed inter-frame space* (DIFS). Note that in IEEE 802.11 based WLANs, the generic *inter-frame space* (IFS) is defined as a minimum duration during which the channel must remain clear following a transmission. The IFS is required so that a station can recognize when a transmission has ended and a potential transmission can start. Once the channel has been sensed idle for DIFS, the station initiates the mandatory *binary exponential backoff* (BEB), also known as *backoff*, before attempting to transmit the packet.

The backoff period guarantees that the channel time is distributed equally between all competing stations and that collisions are minimized. This backoff period is measured in backoff slots of length $\sigma \mu sec$ and is uniformly and randomly selected from $[0, CW-1]$, where CW is the current contention window with the initial value of $CW_{min} = W$. Collision occurs if more than one station transmits during the same slot. If a collision occurs, the contention window of the sender is doubled unless the maximum value $(CW_{max} = 2^m CW_{min},\ m \geq 1)$ has been reached, and the packet is re-queued at the head of line (HoL) of the queue to be retransmitted after another backoff period. The contention window is reset to CW_{min} when the packet has been successfully transmitted or discarded when the retransmission limit $R (R \geq m)$ is reached. A flow chart of the binary exponential backoff process is shown in Fig. 2.15, and in Fig. 2.16 we show the exponential increase in contention window size with a maximum backoff stage of $m = 5$, a maximum retry limit $R = 7$ and an initial minimum contention window of $W = 31$ slots. Note that the contention window is not increased once $W = 2^m CW_{min}$ has been reached.

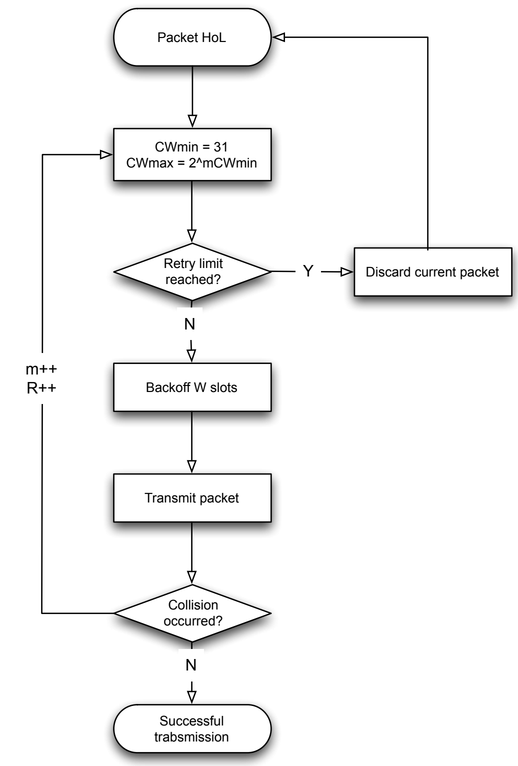

Figure 2.15: Flow chart (algorithm) of the Binary Exponential Backoff (BEB)

Once channel access has been obtained, the station can start to transmit its packet. During the packet transmission, all other nodes will sense the channel

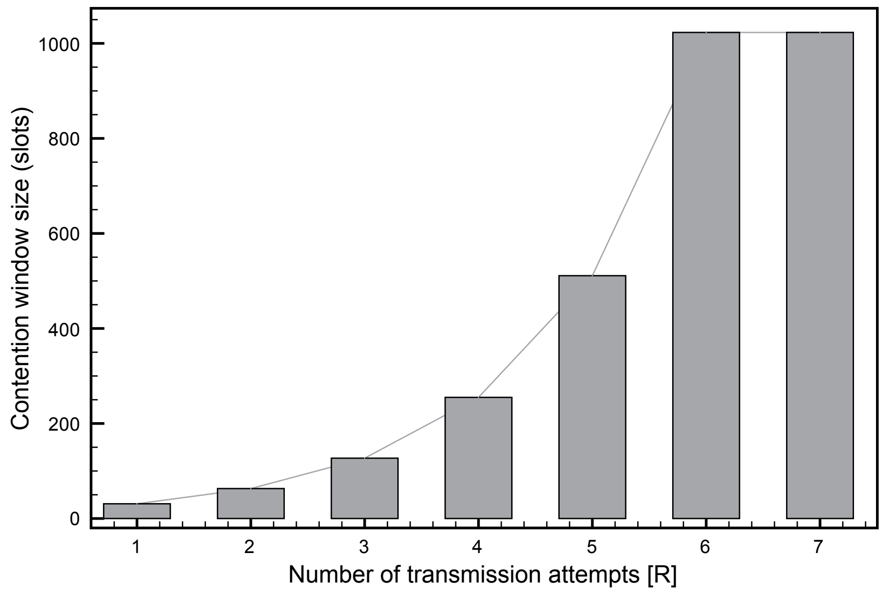

Figure 2.16: Increase in contention window size during the binary exponential backoff

bus
will
nod
The

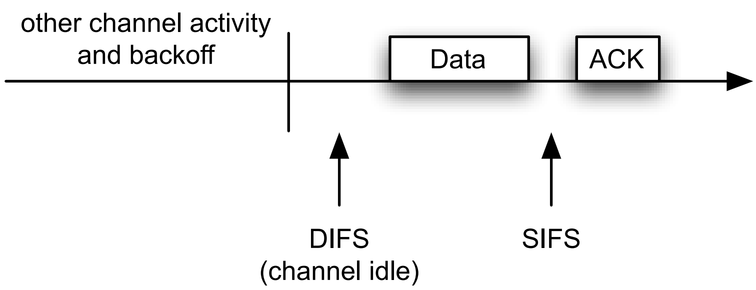

Figure 2.17: IEEE 802.11 DCF without RTS/CTS

To reduce collision frequency in the medium, the RTS/CTS (Request-to-sent/Clear-to-sent) mechanism can be used [85–87]. If the RTS/CTS mechanism is used, the sending node transmits an RTS frame before transmitting the actual data frame. The receiver of the RTS frame will respond by sending a clear-to-send (CTS) frame. This prior exchange of RTS/CTS frames reduces the collision of data packets, because all nodes see the RTS/CTS exchange and thus will not attempt to transmit for a preset time.

The Point Coordination Function (PCF)

Whereas the DCF can be used in infrastructure and ad-hoc WLANs, the point coordination function (PCF) can only be used in infrastructure wireless networks. This is because the PCF relies on a *point coordinator* (PC), which commonly is the access point (AP). The PCF works as follows. The PC transmits a periodic *beacon* frame. The period between these beacon frames is divided into the *contention period* (CP) and the *contention free period* (CFP). During the CP, channel access is according to the the DCF. During the CFP, the PC sends a poll request to each station sequentially to allow them to transmit to the channel. If a node has no packet to send, a timeout occurs and the management node requests a packet from the next node in its polling list.

An example of the operation of the PCF is depicted in Fig. 2.18. In this example we have an AP as the PC and three wireless nodes labeled A, B and C. As shown, after the beacon frame has been transmitted, the PC waits for a duration of a *short inter-frame space* (SIFS) before sending a polling request to node A. After a SIFS, node A transmits its data and ACK frame. The PC then transmits a polling request to node B and also acknowledges the receipt of the frame from node A. In this example, node B has no data to send. Following a PIFS (*PCF inter-frame space*) the PC transmits data and a poll request to node C. Node C has no data to send an only transmit an acknowledge frame to the AP. Finally, the PC transmits a CF-End frame to indicate to all nodes that the conte [illegible] until the n [illegible] ons use D [illegible]

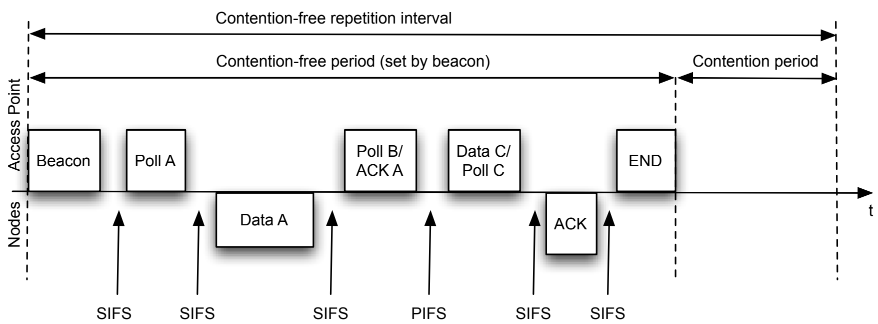

Figure 2.18: The PCF operation

Even though the PCF had been designed for multimedia traffic, studies showed that the associated overhead of polling reduces the overall performance of access [88]. Additionally, the PCF is not mandatory and due to its complexity, the PCF is not widely available in commercial implementations. More details about the PCF can be found in [4, 84] and references therein.

2.2.3 Quality of Service in IEEE 802.11 wireless LAN

While the IEEE 802.11 standard [12] was originally designed to support best effort services in WLAN, IEEE 802.11e [14] was ratified in 2005 to meet the demand of growing volumes of real-time multimedia traffic (such as VoIP) that require some degree of quality of service (QoS). It extends the medium access control of the IEEE 802.11 standard and defines a *hybrid coordination function* (HCF) which provides the *enhanced distributed channel access* (EDCA) and the *HCF controlled channel access* (HCCA). The new standard also allows the adjustment of a number of MAC parameters that were previously fixed. The IEEE 802.11e standard became part of the new specification of the IEEE 802.11 standard in 2007 [4].

The Enhanced Distributed Channel access (EDCA)

The new hybrid coordination function defines the enhanced distributed channel access (EDCA) mechanism to provide contention-based access to the medium. The EDCA in its core functionality is comparable to DCF and uses the CSMA/CA mechanism. This means that a station has to sense the channel as idle for the duration of an *arbitrary inter-frame space* ($AIFS$)[16], followed by the mandatory backoff before the station can attempt to transmit to the channel. However, there is a minor difference in the treatment of the backoff counter between EDCA and DCF. In particular, as shown in [89], the backoff counter in EDCA is resumed before the $AIFS$ elapses, whereas in DCF the counter is reduced the next time the channel is sensed idle. Furthermore, according to the EDCA mechanism, a station has to wait either an additional slot when the medium is idle or an additional $AIFS$ period when the medium is sensed busy, before attempting to transmit, whereas in DCF a station can attempt to transmit whenever the counter has reached zero. We will address this difference in the behavior of the backoff counter later on in our analysis.

To provide QoS, the IEEE 802.11e protocol defines eight user priorities with four different access categories to cater for a range of different traffic types. Each access category uses different and adjustable MAC parameter settings. The first two access categories, *background* (AC_BK) and *best effort* (AC_BE)

[16]The $AIFS$ in EDCA is the equivalent to the $DIFS$ in DCF

Access category	$AIFS$	$TXOPLimit$	CW_{min}	CW_{max}
AC_BK	7	0	31	1023
AC_BE	3	0	31	1023
AC_VI	2	6.016 ms (802.11b) 3.008 ms (802.11a/g)	15	31
AC_VO	2	3.264 ms (802.11b) 1.040 ms (802.11a/g)	7	15

Table 2.4: Default EDCA MAC parameter for IEEE 802.11a/b/g WLANs

can be understood as data traffic access categories, whereas *video* (AC_VI) and *voice* (AC_VO) are specifically designed for real-time multimedia traffic. The four access categories are depicted in Fig. 2.19, and the default EDCA MAC parameter for an IEEE 802.11b and IEEE 802.11a/g are given in Table 2.4. Note that the values for $AIFS$, CW_{min} and CW_{max} in Table 2.4 are given in

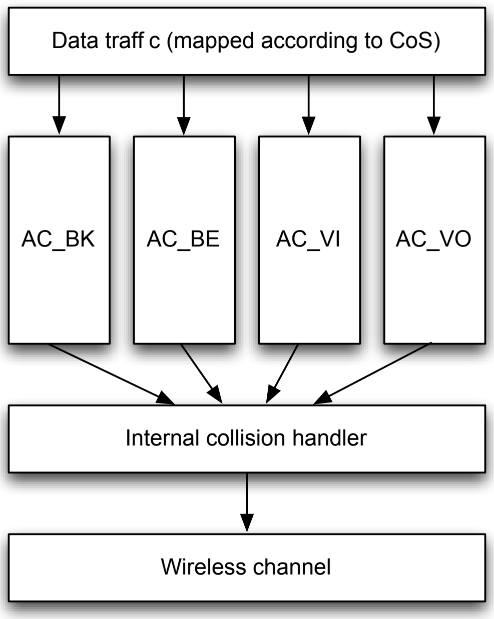

Figure 2.19: EDCA access categories

The $AIFS$ (arbitrary inter-frame space) parameter of the IEEE 802.11 protocol is comparable to DIFS in DCF and defines the time period the channel has to be sensed idle before a node can attempt to transmit. By adjusting the $AIFS$ parameter the system can control which traffic type can access the channel, before any other traffic type can attempt to use the channel. This is because a shorter $AIFS$ period allows an earlier channel access.

The $TXOPLimit$ parameter specifies a contention-free duration during which a TXOP-holder has uninterrupted access to the channel, thus allowing the continuous transmission of multiple packets without re-contending for the channel as shown in Fig. 2.20. To avoid contention from other nodes during the time duration of $TXOPLimit$, the TXOP holder is allowed to commence its transmission of a consecutive packet after a short inter-frame space (SIFS) following the completion of the immediately preceding frame exchange sequence, i.e. on receipt of an acknowledge (ACK) frame.

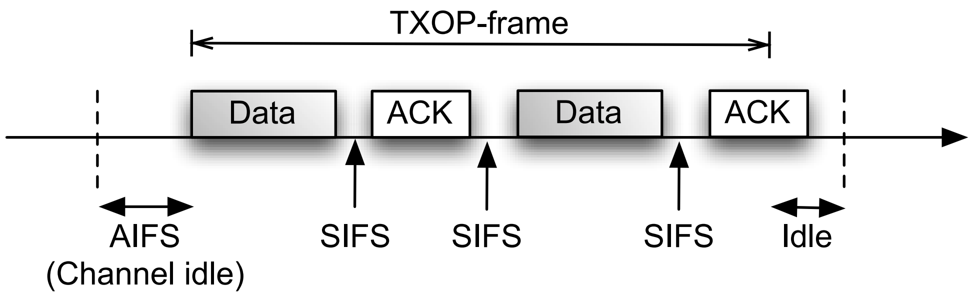

Figure 2.20: Transmission of multiple packets during the $TXOPLimit$ period

The contention window parameter CW_{min} and CW_{max} control the frequency with which a station can access the channel. Recall that a station has to wait

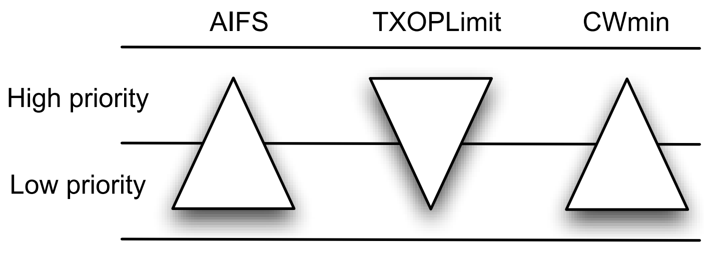

Figure 2.21: IEEE 802.11e MAC parameter indication of high and low priority

As shown in Figure 2.19, before a packet is sent, an internal collision resolution mechanism resolves any *internal collision*. An internal collision can occur if the backoff counter of the mandatory backoff for packets in multiple access queues reaches zero, and henceforth these queues at a station attempt to transmit at the same instance. In case of an internal collision, the internal collision resolution mechanism will grant channel access to the packet of the higher access category. The packet of the lower access category is then scheduled for retransmission after an additional backoff process. The different settings of

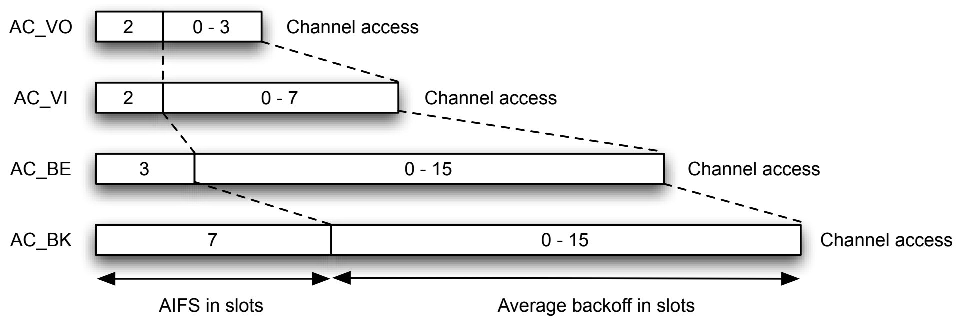

Figure 2.22: EDCA channel access for the four access categories

The HCF controlled channel access (HCCA)

The HCF of the 802.11e protocol also specifies HCF controlled channel access (HCCA), which, similar to PCF, is designed for the contention free channel access. The main difference between HCCA and PCF is that the time period between the beacon frames in PCF is strictly divided into CFP and CP, whereas in HCCA, the PC can initiate a CFP an any time during a CP. In this case, the PC initiate the *controlled access phase* (CAP) whenever the PC wants to transmit to or receive a packet of a specific station. During the CP all nodes work according to EDCA. In contrast to PCF, HCCA also introduced the notion of traffic classes and traffic streams. Each node can communicate its queue saturation of a traffic class to the PC, which can then assign a different priority to that node. Due to its fine grained ability to control QoS, HCCA is a superior access function. However, similar to PCF, its implementation in not compulsory and due to its complexity, there are no commercial implementations available. More details about HCCA can be found in [90, 91].

2.3 VoIP in IEEE 802.11 WLAN

2.3.1 Introduction

At the beginning of this chapter, we have outlined that many home and business users now use VoIP to lower their communication costs. In particular software VoIP applications such as Skype or Google Talk can provide significant savings because calls between Skype or Google Talk users are free of charge. Furthermore, with the integration of voice and video features into social media services like Facebook[17] or Google+[18] it is possible that traditional voice communication based on the public switched telephone network (PSTN) will become redundant in the near future.

Further enhancements to the IEEE 802.11 protocol to support QoS and the reduction in cost of wireless technology has led to an integration of wireless capabilities into mobile devices such as laptops, PDAs and cell phones. In particular the use of mobile devices has seen an increased demand for wireless access. As highlighted in Chapter 1 it is expected that the demand on mobile data traffic will increase by 92% annually between 2010 and 2015 [5].

With an increase in computing capabilities and memory in cell phones, voice over IP applications, e.g. Skype or Google Talk are now readily available for smart-phones such as Apple's iPhone or the Android based HTC. Wireless VoIP is an important emerging service due to the potential to replace the current cell phone communication wherever a WLAN is available. In Fig. 2.23[19] we show an overview map of available AT&T wireless hotspot locations. As shown, AT&T hotspots are available in a variety of locations. For example, there are more than 4000 AT&T hotspots in California alone, in places such as Starbucks, McDonalds restaurants or airports. The T-Mobile service of the Deutsche Telekom for example lists more than 45000 hotspots in over 20 countries[20]. Similar maps are available for a variety of other wireless providers around the world. The examples outlined above show that wireless networks are now becoming ubiquitous and that wireless voice over IP can now challenge traditional cell phones services based on GSM, 3G or 4G technology, as it can deliver similar cost savings to its wired counterpart.

[17] http://www.facebook.com
[18] http://www.google.com/+
[19] http://www.att.com/gen/general?pid=13540
[20] https://selfcare.hotspot.t-mobile.com/locations/viewGlobalLocations.do

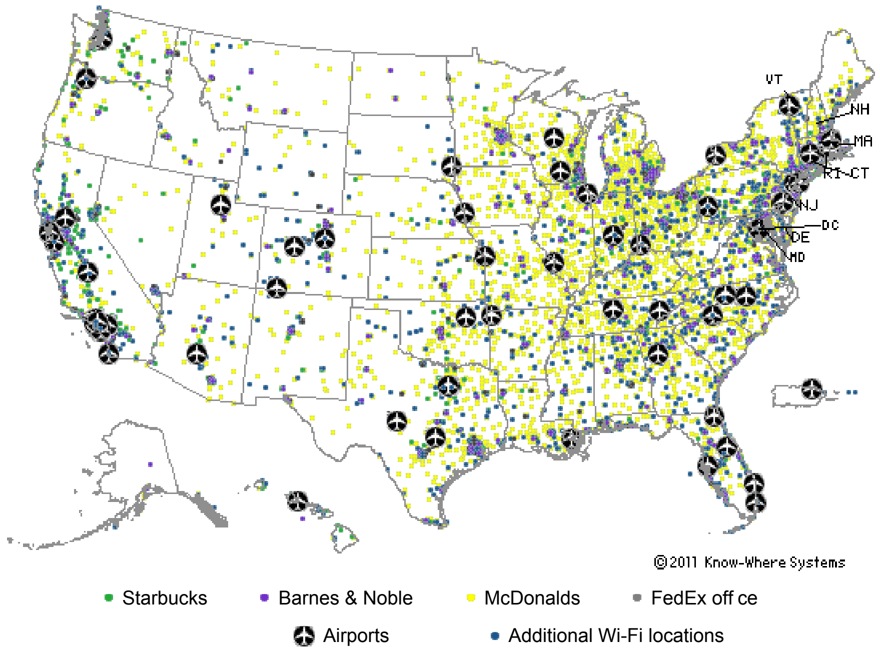

Figure 2.23: AT&T wireless hotspot locations in the United States

2.3.2 Wireless VoIP performance

Earlier in this chapter, we argued that wireless voice over IP has the potential to replace today's mobile phone communication, as it can provide real cost savings to individuals and businesses. However, to make VoIP in WLANs a viable alternative, a variety of challenges need to be overcome so that high user satisfaction is guaranteed. For example, it is known that the wireless medium is unreliable and suffers from interference [92, 93], because an IEEE 802.11b/g/n WLAN uses the same frequency (2.4 GHz) as other common radio devices such as cordless phones or bluetooth devices [94–97]. Furthermore, fading [98] or mobility introduce a new level of complexity, as for example, a user may roam between multiple access points (APs) and require handoff [99,100], which also affects VoIP quality [101]. Although these are open issues that need to be addressed, one of the major concerns for wireless VoIP is the limited number of VoIP calls that can be supported with an acceptable level of quality.

For example a G.729 voice codec generates an 8 kbit/s voice stream and a G.711 voice codec generates a 64 kbit/s voice stream, respectively. As voice calls are full-duplex, a single call requires a total bandwidth of approximately 16kbit/s or 128kbit/ using the above codecs. Therefore, based purely on bit rate, an IEEE 802.11b WLAN for example would be expected to support $11/0.015 \approx 733$ calls using a G.729 or $11/0.128 \approx 85$ voice calls using a G.711 voice codec.

Similarly, an 802.11a/g WLAN should be able to carry up to $54/0.015 \approx 3600$ calls or $54/0.128 \approx 421$ voice calls using the G.729 and G.711 voice codecs, respectively. However, the actual VoIP capacity is less than 10% of the expected number of VoIP calls [17,73,102,103]. As the required (aggregated) bandwidth of VoIP calls is well below the available bandwidth, as shown in Fig. 2.24, suggests that the bandwidth is not limiting the number of supported VoIP calls in a wireless LAN.

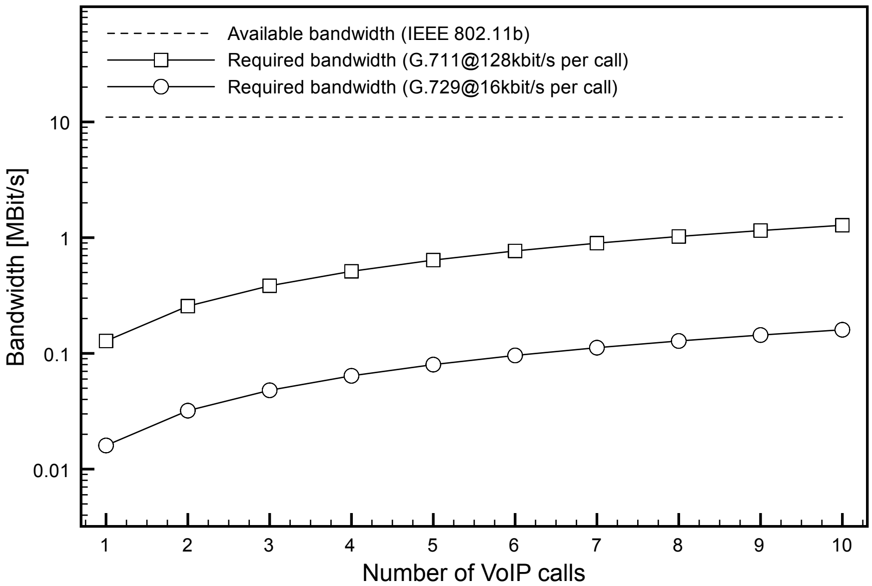

Figure 2.24: Available bandwidth vs. required bandwidth for full-duplex VoIP calls

The significant difference in expected and actually observed number of VoIP calls has attracted considerable research effort. Research focusing on the VoIP capacity only [2, 26, 27, 64, 65, 73, 90, 102–105] has identified that the access point (AP) in an IEEE 802.11 infrastructure WLAN is a bottleneck and limits the number of calls. This is because in an infrastructure WLAN, all traffic from and to the wireless nodes has to pass through the common AP, acting as a bridge between the wired and the wireless network. Due to the design of the medium access control, each station has an equal probability of obtaining channel access. Therefore, with an increasing number of wireless nodes in the WLAN, the probability of the AP winning the channel contention decreases, because the AP has to compete with the wireless nodes to access the channel for each packet on the downlink. It can be shown that with an increasing number $N\epsilon$ of wireless nodes the channel access probability of the AP is $1/(N\epsilon+1)$ whereas the channel access probability of the wireless nodes is $N/(N\epsilon+1)$ [1,2]. As shown in Fig. 2.25 the channel access probability of the AP with increasing $N\epsilon$ rapidly decreases. Due to the low channel access probability of the AP, packets arriving at the AP will be queued. With an increasing number of wireless nodes the number of arriving packets is increasing and at some stage packet loss at the AP will occur due to a limited buffer space. Also, due to the buffering at the AP, the user perceived ear-to-mouth delay increases, and at some stage reaches a level that is deemed to be unacceptable. The fundamental problem is that the AP is the bottleneck in WLAN, limiting the number of voice calls that can be supported in an IEEE 802.11 infrastructure WLAN.

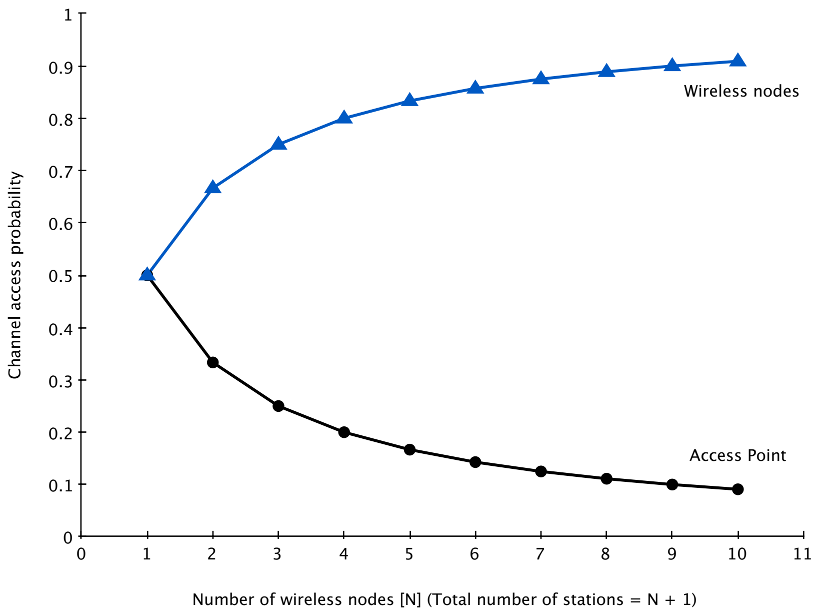

Figure 2.25: Channel access probability of the AP and the N wireless nodes [1, 2]

VoIP capacity analysis

There is a variety of literature that discusses the VoIP capacity in IEEE 802.11 WLANs [26, 27, 64, 65, 73, 101–104, 106–117] and references therein. The approach on assessing VoIP capacity ranges from analytical modeling [26, 65, 73, 102–104] to real testbed measurements [27, 108–111]. The majority of the work in this area focuses on IEEE 802.11 infrastructure WLANs, where the channel access is controlled by the DCF, and only a few consider EDCA when investigating the VoIP capacity in a WLAN. An EDCA voice capacity is commonly considered, if the channel is shared with other types of traffic, e.g. TCP, or if the adjustable MAC parameters are used to improve the VoIP capacity, both of which we discuss later on. As the majority of wireless networks are based on infrastructure WLANs where the channel access is controlled using DCF or EDCA, our discussion concerning the VoIP capacity is focused on such networks. A discussion of VoIP in PCF controlled WLANs can be found in [113, 114], and works in [111, 115–117] provide a VoIP capacity analysis for single and multi-hop ad-hoc wireless networks.

Using analytical models to determine the maximum number of VoIP calls is used in [26, 65, 73, 102–104]. Even though all of these research studies develop an analytical model, the approaches used differ. In [65, 73] Hole and Tobagi for example use a simple approximation equation to determine the number of VoIP calls that can be maintained. In particular, the capacity is derived based on the fraction of time the channel is occupied by the VoIP calls. They show that the number of VoIP calls (C) is calculated by[21]

$$C\epsilon = \frac{1}{2}\left(\frac{(1/T_s)}{\lambda\epsilon}\right)$$

where T_s is the time required to transmit a single packet, $\lambda\epsilon$is the arrival rate per half-duplex (one-way) call in packets, and the $1/2$ factor is required as the total number of packets in the uplink and the downlink direction is equal. For example, if $T_s = 500\mu s\epsilon$and a half-duplex VoIP call generates $\lambda\epsilon = 100$ packets per second, the WLAN can support up to 10 voice calls. Using this approach, Hole and Tobagi [73] show that an IEEE 802.11b WLAN can support 6 CBR voice calls only. Medepalli et al. [65] on the other hand consider VBR voice traffic and show that in this scenario the WLAN can then maintain up to 11 voice calls. The difference in call capacity is due to the different traffic intensity. As discussed in Section 2.1.3, a VBR voice stream, on average, consists of fewer packets, thus reducing the level contention in the WLAN that subsequently increases the number of calls that can be supported.

The authors of [102, 103] use a different analytical approach to determine the VoIP capacity in an IEEE 802.11 infrastructure WLAN. Even though the metrics used to obtain the VoIP capacity in [102, 103] are different, both analytical models use a *mean-based* approach, whereby the capacity is calculated using average values of the arrival-, the service- or the collision time of packets. Using the fixed-point formulation to obtain the average service time, Hedge et al. [102] apply the Pollaczek-Khinchin formula [118] for the mean queueing time to derive the delay at the AP, that is the total delay a packet experiences in

[21] Note that the notation has been adjusted for simplicity. Please refer to [65] and [73] for their actual notation used.

the downlink direction. Using the analytical model, Hedge et al. show that with an increasing number of VoIP calls in the WLAN, the delay at the AP is increasing, and at some stage exceeds a maximum delay bound, in their model 100 ms, before the voice call quality is no longer considered acceptable. In [103] the authors also use a fixed-point formulation, however, instead of using network delay to determine the maximum number of VoIP calls, Cai et al. [103] use a throughput model. In this model the VoIP capacity is determined by the queue stability condition, that is the arrival time (λ) does not exceed the service time (μ). In their approach it is shown that with an increasing number of VoIP calls, the average service time increases, and as a consequence the service rate decreases as shown in Fig. 2.26.

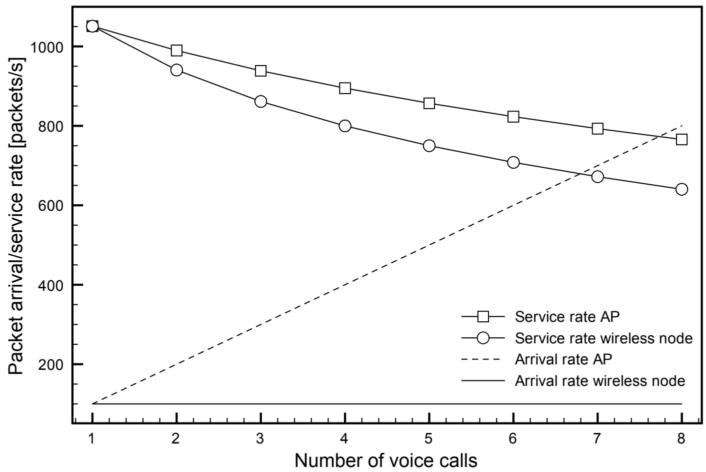

Figure 2.26: Arrival and service rate of a G.729 voice codec with a 10 ms sampling period in an IEEE 802.11b WLAN

The results reported in [102, 103] are similar, and it is shown that an IEEE 802.11b WLAN can support only 8 [102] or 6 [103] voice calls using a G.711 voice codec with a 10 ms sampling rate. Although the methods to obtain the VoIP capacity in [102, 103] and [65, 73] are different, the results reported are comparable. Nevertheless, the analytical methods used in [102, 103] provide an insight into the VoIP capacity limit, as a variety of internal metrics, such as the collision or attempt probability, can be derived. This insight can be used to develop different solutions to increase the number of VoIP calls that can be maintained. Note that Hedge et al. extend their analysis to investigate the VoIP capacity when the channel is shared with downlink data traffic, and to study if changes to the minimum contention window and the retry limit leads to an improved VoIP capacity. Setting a smaller than default value for the contention window allows an increased frequency with which a station can attempt to access the channel. This is because the random backoff is drawn from a smaller contention window and as a result the overall duration of the backoff period is reduced. Also, setting a smaller retry limit at a station lowers the maximum contention window size and allows a packet to be discarded earlier if multiple collisions are experienced. By discarding a packet after only a few collisions, the delay experienced by subsequent packets in the queue is reduced, which may increase the VoIP capacity, depending on the tolerable delay.

A similar approach to [102] is used in [105], where the authors also use a network delay to determine the VoIP capacity. Furthermore, Oh et al. also consider changes to the minimum contention window and the retry limit. However, the main difference between [102] and [105] is the analytical model. Hedge et al. used a mean-based approach based on an $M/G/1$ queueing system, whereas the authors in [105] used a Markov-chain model that was derived from the popular Bianchi approach [119]. In [119] Bianchi developed a two dimensional Markov model to capture the backoff behavior of a single node in a DCF controlled IEEE 802.11 WLAN to study the saturated throughput that can be achieved in these networks. Based on Bianchi's closed-form expression for the saturated throughput, a variety of works have adopted this approach and suggested modification to the model to capture a variety of other network conditions such as unsaturated nodes, or an IEEE 802.11 EDCA WLAN. Oh et al. follow the Bianchi approach and modify the model to suit an EDCA controlled WLAN. Using the model and simulation, Oh et al. show that the wireless network can maintain 11 voice calls using a G.711 voice codec with a 20 ms sampling rate without changes to the parameter, and the results are in line with those reported in [26] where the authors also use a Markov-chain model. Using their model, similar to the work in [102], Oh et al. then study if changes to the contention window and the retry limit will improve the VoIP capacity. It is shown that a VoIP capacity increase to 17 voice calls can be achieved when $CW_{min} = 3$ and the retry limit is set to $R\epsilon = 10$. Markov-chain models have also been used in [26, 104, 120–123]. In [122, 123], Gao et al. follow a similar analysis to [105] and [102] to improve the performance in these WLANs by setting different values for the minimum contention window size, and the results are comparable to those reported in the above literature.

The research results presented above highlight that the AP is a bottleneck in an IEEE 802.11 infrastructure WLAN, and that the limited channel access of the AP rather than the available bandwidth of the wireless channel imposes the

VoIP capacity restriction. Therefore one could argue that ad-hoc WLANs are more suitable for VoIP, because in an ad-hoc WLAN there is no AP bottleneck. Because ad-hoc WLANs have been designed for different scenarios, as we have discussed in Section 2.1.1, ad-hoc WLANs are commonly not used for voice over IP communication, as they are limited to *internal*[22] communication only or maybe used in mesh networks. Nevertheless, research in this area has shown that the number of VoIP calls depends on the topology and the distance between the wireless nodes. More details about voice over IP in ad-hoc WLANs can be found in [110, 115, 116, 124–127].

VoIP capacity improvements

Most of the research discussed in the previous section provides an analysis of the VoIP capacity in an IEEE 802.11 infrastructure WLAN. In this section we discuss some of the solutions proposed to improve the VoIP capacity in WLAN. There has been considerable research investigations discussing a variety of solutions to tackle the limited VoIP capacity in WLAN. The literature in this area can be split into two areas, namely solutions that address the limited VoIP capacity explicitly [2, 66, 105, 128–132], and solutions that improve the overall performance of the WLAN, and as a result have the potential to improve the number of voice calls that can be supported [129, 133–135]. Most of the proposed solutions either consider a DCF or EDCA channel channel access. However, [88, 136] consider the use of PCF, and [125, 126] additionally consider ad-hoc WLANs.

The solution presented in [128, 132] suggests a *multiplex-multicast* (M-M) scheme, whereby VoIP calls are multiplexed for inter-domain[23] transfer, and within the destination wireless network the voice calls are multicast in the WLAN to reduced the traffic in the downlink direction. By using multicast in the downlink direction, the number of streams is reduced from $2N\epsilon$to $N\epsilon+1$ streams, assuming $N\epsilon$full-duplex VoIP calls. To show the increase, Wang et al. use a similar analytical approach as discussed in [65, 73]. Wang et al. [128] show that the proposed solution can increase the VoIP capacity and that for example an IEEE 802.11b WLAN can maintain 17 instead of 10 voice calls using a G.711 voice codec with a 20 ms sampling rate. The authors also extend their work and show that a similar increase in capacity can be achieved if an IEEE

[22] Internal communication here refers to VoIP communication between the ad-hoc wireless nodes only, without an interaction with nodes external to the WLAN

[23] An *inter-domain* transfer here is a transfer between two separate VoIP networks

802.11a/g WLAN with a 54 Mbit/s data rate is considered. Even though the proposed solution significantly increases the number of VoIP calls, it requires changes to the protocol, additional hardware for the inter-domain transfer as well as the WLAN to be multicast enabled, which may not be desirable. A similar solutions using the multiplex-multicast approach is also presented in [137].

Other solutions [127, 129, 138–140] and reference therein, suggest the modification of the *call admission control* (CAC) [141, 142]. The CAC mechanism in an IEEE 802.11 WLAN regulates admissions of new streams to the WLAN. If a new stream wants to join, it requests required resources. If the resources are available, the stream is admitted, and rejected otherwise. This provides a further protection for real-time traffic, and the CAC mechanism complements the QoS control. In [129], the authors propose a novel call admission control mechanism referred to as *adaptive transmission interval call admission control* (ATICAC) that does not require any additional hardware. The novel scheme works such that the access point (AP) in the WLAN calculates an estimate for the collision probability based on a channel *busyness* indicator (see Eq. (15) in [129]) whenever a new call attempts admission. If the estimate for the collision probability is below 0.1, the new call is admitted as the WLAN is not saturated. If the estimate is above 0.1, the AP can either reject the new call, or adjust the transmission interval of all VoIP calls, including the new call. By adjusting the transmission interval the mechanism effectively changes the sampling rate of the voice codec such that fewer, but larger packets are transmitted in the downlink direction. The authors argue that this scheme doubles the number of VoIP calls that can be supported with only a minor increase in call delay. However, this solution may not be desirable, because it requires significant changes to the protocol, and furthermore, the AP requires additional software to perform the frame aggregation on the AP to increase the transmission interval.

The work in [138, 143, 144] also suggest different call admission control mechanism to improve the VoIP performance. In a similar fashion to the changes of the transmission interval, Jeong et al [131] suggests to either use ACK aggregation, frame aggregation or link adaptation to boost the VoIP capacity in an IEEE 802.11 WLAN. Using an analytical model based on the Bianchi approach, Jeong et al. show that sending a single ACK to acknowledge multiple frames (ACK aggregation) will only increase the VoIP capacity marginally, that is two VoIP calls if no acknowledgement is sent at all. The change in channel rate (link adaptation) will also not lead to an increased VoIP capacity. In particular, VoIP capacity is reduced from 5 to 4, 3 and 2 calls, if the channel rate is reduced from 11 Mbit/s to 5.5 MBit/s, 2 MBit/s or 1 MBit/s, respectively. Frame aggregation however, can provide a capacity gain of up to 300% in terms of number of VoIP calls that can be supported. Frame aggregation, is the transmission of one large frame and/or multiple frames per channel access. Frame aggregation results in a reduced overhead, as a station does not need to re-content for the channel for each frame. We utilized a similar technique in Chapter 3 and show the benefits of this scheme in terms of a gain in the number of VoIP calls. Even though the proposed solution provides a significant increase in VoIP capacity ($\approx 300\%$), the modification required to achieve this increase is significant, and as such may only be applicable to a highly customized wireless networks. Nevertheless, different frame and ACK aggregation methods are part of the IEEE 802.11n protocol [22] and it is yet to be investigated which aggregation technique provides the maximum VoIP capacity.

Alternative solutions based on the IEEE 802.11 QoS mechanism have been proposed in [2, 64, 105, 123, 125, 130, 133, 145–147]. Whereas some solutions utilize the QoS mechanism introduced in the IEEE 802.11e protocol [14], others suggest modifications of the DCF channel access mechanism to achieve equal results [64, 147].

Recall, that the IEEE 802.11 QoS mechanism allows the modification of the contention windows (CW_{min}, CW_{max}), the $AIFS$ and the $TXOPLimit$ parameter, and therefore different priorities can be assigned to different traffic types. Moreover, different settings can also be applied at a wireless node and the AP.

The works in [123,125,145] suggest modification of the contention window parameter to increase the number of VoIP calls. By setting a smaller than default value of the minimum contention window CW_{min}, a station can attempt to access the channel more frequently. Due to the increased channel access, the total number of packets queued at the AP is reduced and as a consequence VoIP capacity is increased. Whereas the work in [123] is only concerned with VoIP capacity, the authors in [125] and [145] also consider the case where the channel is shared with data traffic. In both works, the authors use the arbitrary inter-frame space (AIFS) parameter to distinguish between VoIP and data flows. In particular, to provide a higher priority, the VoIP flows use a smaller $AIFS$ parameter. The results in both works show that using $AIFS$ in a multi-traffic environment is effective such that it protects the VoIP flows in terms of the VoIP capacity. Note that in Chapter 6 we will provide a more detailed discussion about the impact of data streams. The increase in VoIP capacity as reported in [123] by setting a smaller than default CW_{min} parameter is approximately 30%, e.g. the VoIP capacity is increased from 11 to 14 voice calls using a G.711 voice codec. These results are in line with those reported in [125, 145, 146].

Changes to the $TXOPLimit$ parameter have explicitly been studied in [130] and have been extended in [2]. Recall that the $TXOPLimit$ parameter specifies a duration during which a station that gains channel access has uninterrupted access to the channel. This means that a station can either transmit a large packet, or multiple small packets, until the $TXOPLimit$ timer lapses. In both works, the $TXOPLimit$ parameter is assigned to the AP only and is set such that the AP can transmit multiple packets during the $TXOPLimit$ duration. By setting an increased value of $TXOPLimit$ at the AP results, on average, in a lower queue utilization at the AP, and therefore an increased VoIP capacity, as packet loss or the violation of the queue stability condition can be delayed. Whereas the work in [130] uses a modified Bianchi model to obtain the number of voice calls that can be maintained, Dangerfield et al. [2] use real testbed measurements to derive the VoIP capacity. Based on their results, Dangerfield et al. suggest that the $TXOPLimit$ value in terms of packets per channel access should be set equal to the number of VoIP calls in the WLAN, or halved, if also the default contention window size is reduced by 50%. However, as we will show later in Chapter 3 the $TXOPLimit$ parameter setting is too aggressive and will impact the overall system performance. Furthermore, it requires a priori knowledge of the number of VoIP calls in the WLAN. A similar suggestion in [143] for the ideal setting of the $TXOPLimit$ parameter is also too aggressive and will also impact the overall system performance.

Even though the research described above shows that setting an increased value of $TXOPLimit$ parameter can improve the VoIP capacity, their analysis overestimates the ideal value of the $TXOPLimit$ parameter and does not take into account any impact on the wireless voice nodes. In Chapter 3 we will show that we can obtain a similar increase in VoIP capacity as reported in [2], but with a much lower value of the $TXOPLimit$ parameter. We will show that our setting of the $TXOPLimit$ parameter is optimal, such that the maximum VoIP capacity is achieved, and that further increasing the $TXOPLimit$ parameter value provides no improvement of the capacity and impacts on the overall system performance. We will confirm our findings using a detailed analytical model, simulation and real testbed measurements. We also investigate if a further capacity gain can be achieved in conjunction with changes to the CW_{min} and $AIFS$ parameter. Finally, we will also show that our findings hold for variable bit rate voice flows, which have not been discussed in [2, 130, 143].

Buffer requirements for VoIP calls

In the previous sections we have highlighted some of the research investigations into limited VoIP capacity and means to improve the number of voice calls that can be supported in an IEEE 802.11 WLAN. As outlined, there are different ways of determining the voice capacity, for example, using an analytical model [102, 103] or testbed measurements [27, 65]. However, an important aspect of the VoIP performance has not been discussed in the literature - the buffer requirements for VoIP calls. There is extensive literature discussing buffer size requirements in wired networks [148–157] and references therein, but only a few works discuss buffering requirements in IEEE 802.11 WLANs [2, 25, 158], and in particular the buffering requirements in WLAN for VoIP traffic [25, 109]. Investigating the buffering requirements is important, as one might argue that increasing the buffer size at the AP will improve the number of VoIP calls that can be supported, because more packets can be queued and, as a result, packet loss may not occur, or may be delayed such that an increased number of voice calls can be maintained.

For example, in [158], the authors study the buffering requirements for TCP flows and argue that the common *Bandwidth-Delay Product* (BDP) [159] cannot be used to adequately provision the buffer requirements in a WLAN. This is because of the interaction between the congestion control and the contention based channel access that can lead to underutilization of the link. Instead of using the BDP, Li et al. [158] suggest adjusting the buffer depending on the network conditions such as the number of flows and the level of contention. A similar suggestion is made in [25], where the authors argue that the buffer size at the AP should be set at least equal to the number of voice calls. A further investigation of [25] in [2] shows that increased throughput can be achieved if the AP has a buffer of 30 packets, however, Dangerfield et al. [2] conclude that the VoIP capacity is independent of the buffer size at the wireless nodes, but a voice capacity gain can be achieved with an increased buffer at the AP, i.e. a capacity increase from 10 to 12 voice calls. Nevertheless, the authors in [2] argue similarly to [25] and conclude that the buffer size at the AP should be proportional to the number of voice calls in the WLAN. In Chapter 4 we will provide a different insight into the buffering requirements for VoIP calls based on our analytical model, simulation and testbed measurements. In particular we will show that there exists a minimum buffer size at the AP that maximizes the VoIP capacity. Even though our findings for the buffering requirements for the wireless nodes is inline with those presented in [2, 25], we will also show that the bottleneck shifts from the AP to the wireless nodes, caused by too large values of the IEEE 802.11 QoS parameter, can be alleviated with larger buffer sizes at a node. Furthermore, our findings provide strong evidence that the proportional buffer size is not required, because we show that the number of VoIP calls that can be supported is independent of the buffer size at the AP, given the minimum buffer size outlined above. Using the insight gained from our analysis, we will also propose a novel VoIP approximation equation. This equation allows us to obtain a better understanding of VoIP capacity when a channel access priority is assigned to the AP using the $TeXOPeLimite$parameter.

Dynamic voice codecs in WLANs

A recent development in VoIP is the use of *dynamic* voice codecs which are designed to adapt to changes in network condition. For example, SILK [58,160] used in Skype V.4 or SPEEX [161] used in Google Talk, monitor the call quality and adjust the codec parameters accordingly. The goal is to maintain the call with reduced quality during congestion periods in the network. In this context mixed codecs can be seen in some of the previous work. In [26] Harsha et al. consider two types of voice traffic using different codecs. In their work it is assumed that both types of voice traffic share the channel and are served by the AP without traffic differentiation. Also in [162] a dynamic adaptation of the voice codec is proposed to suit the change in transmission rate of the WLAN.

Here we propose a novel scheme based on the IEEE 802.11 quality of service (QoS) mechanism to exploit the tradeoff between the codec quality and priority to maintain the call during periods of high contention without compromising the individual call quality. At the same time the proposed scheme also improves the overall number of calls that can be supported by the WLAN. The novelty in our approach is that our scheme gives an incentive to users who are willing to use a lower quality codec and thus reduce the overall contention when there is a high traffic load in the medium. The incentive is implemented by giving priority at the AP to traffic originating from lower quality codec users. Note that this scheme can be easily implemented because the priority is only applied at the AP where the network bottleneck is located. Also, users can voluntarily choose the codec by monitoring their own call quality as is done in Skype.

In order to understand the benefits of the above proposed scheme, we develop a detailed analytical model. Based on our analytical model that we develop in Chapter 3. In particular, we extend the existing model to accommodate multi-codec voice streams and to include an internal collision caused by the use of priority at the AP. Note that in [120] and [145] the authors investigated the use of multiple queues based on the IEEE 802.11e protocol, however, they did not consider the effect of internal collisions. We will show through our analytical model that the impact of this internal collision cannot be neglected.

We assess the quality of calls in the WLAN using the proposed scheme using the ITU-T E-model [19] as discussed in Chapter 2. Note that although the analytical model has been developed to analyze the scenario outlined in Section 5.1, the proposed model is sufficiently versatile to enable study of a range of scenarios, such as the interactions between voice and video traffic in a WLAN using the IEEE 802.11 QoS mechanism. Additionally, the model allows the investigation of a range of network parameters such as collision probability and queue utilization at a station and allows different network metrics, such as throughput, to be derived. The model is also flexible enough to be extended to incorporate other traffic types, such as traffic generated by TCP connections.

Concurrent VoIP and TCP traffic in IEEE 802.11 WLANs

There is a variety of literature that investigates the performance of an IEEE 802.11 WLAN when the wireless channel carries VoIP and TCP data traffic [20, 21, 90, 102, 120, 133, 136, 147, 163–168] where channel access is controlled either using DCF or EDCA. Whereas in the previous section, the literature was mainly concerned with VoIP flows only, in real networks this is unlikely, and the VoIP flows have to compete for the channel with concurrent TCP data flows, such as Web traffic or Email. Due to the increased level of contention in the WLAN, it is anticipated that the overall number of VoIP calls is further reduced. This drop in capacity is shown in [163], where Bellalta et al. show that the aggregated VoIP throughput is slightly reduced when the channel is shared with downlink TCP flows, such as a file download. However, Bellalta et al. [163] also show that the throughput is significantly reduced if the channel is shared with uplink TCP flows. We will discuss the reason behind the difference later in Chapter 6. Bellalta et al. also propose a new channel access mechanism using the IEEE 802.11e QoS mechanism to improve the throughput, whereby a smaller contentions window is set at the AP. Similar results in terms of aggregated TCP and VoIP throughput are shown in [147]. In [147] Kim et al. also provide a delay analysis if the channel access uses DCF and EDCA. Furthermore, in [147] it is proposed to use a so-called *vDCF* (voice DCF), which is based on piggybacking and packet bursting. Lucani et al [133] show that with an increasing number of web users generating TCP traffic, the number of VoIP calls in an 11 MBit/s WLAN decreases from 6 to approximately 4 voice calls.

An extensive analysis based on a Markov based analytical model is presented in [90, 104, 120, 169], where the authors consider a HCCA channel access for VoIP and TCP data traffic is served by EDCA. Using network delay as a QoS measure to derive the number of VoIP calls, Harsha et al. [90] show that the WLAN can only support 7 voice calls if the TCP flows require 2 MBit/s of bandwidth. The results for TCP throughput are in line with results presented in [20, 167] who do not consider voice flows.

In [170] the authors investigate by simulation the impact of uplink and downlink TCP flows on the experienced delay for VoIP traffic. Yu et al. [170] first evaluate the performance of VoIP traffic only, and the results presented are comparable to those presented in the discussed literature in the previous sections. The authors then evaluate the difference in VoIP delay when the AP uses a single and a dual-queue, whereby the two queues are used to transmit a single type of traffic only. The results for a single queue are equal to those presented in [163], however, using a dual-queue approach, it is shown that the VoIP performance is increased as the experienced VoIP delay is significantly reduced.

In a recent study by Brouzioutis et al. [165], the authors develop an analytical model for the IEEE 802.11 DCF when the channel is shared between VoIP and TCP data traffic. The modeling approach is based on the common Bianchi model [119], but extended such that it can capture both traffic types. Using their model to derive delay, jitter and VoIP packet loss, it is shown that the number of VoIP calls is decreasing by two calls for each single data flow. For example, Brouzioutis et al. show that the WLAN can support up to 12 voice calls when the network delay is used to derive the call capacity, and the channel is used for VoIP only. However, when the channel is shared with 1, 2, 3 and 4 data streams, the VoIP capacity is reduced to 10, 8, 7 and 5 voice calls, respectively. The authors argue that the reduction by approximately two calls per data stream is caused by the large payload of the TCP traffic and the small RTS/CTS exchange prior to the transmission of TCP data traffic.

Even though results have shown that TCP data traffic reduces the VoIP capacity in an IEEE 802.11 WLAN, there are different factors that need to be considered. For example, in [163] the authors consider the difference in TCP flow direction, however, the impact of different TCP window sizes is neglected.

In [170] the authors utilize a dual-queue approach similar to two different access categories of the IEEE 802.11e protocol, and they show that this approach provides some protection for the VoIP flows in terms of lower delay, however, the flow direction is not considered. The work in [145] considered MAC parameter tuning for VoIP and data traffic, and for example it is shown that a higher VoIP performance can be achieved when data traffic is present, if a smaller contention window is set for VoIP traffic only. However, Narbutt et al. [145] consider an UDP downlink stream as data traffic rather than an actual TCP stream. The authors argue that the benefits of using a UDP rather than a TCP data stream is threefold; namely that a UDP data flow will more accurately estimate the traffic load, that using UDP will constitute an upper bound for the throughout possible with TCP, and that the retransmission of lost packets is managed by the MAC rather than the TCP mechanism. However, by limiting the data traffic to UDP flows only does not take the interactions between the voice and TCP packets and the MAC and the TCP congestion control into account.

In our analysis in Chapter 6, we will investigate by simulation the impact of uplink and downlink TCP flows on the VoIP capacity. This allows us to study the impact of the flow direction and, in particular, the interactions of the TCP and VoIP flows in either direction. Based on our initial finding, we will follow the approach by [145] and also consider MAC parameter tuning to increase the VoIP capacity and maximize the TCP throughput. However, based on our findings in Chapter 3 we will focus on the $TXOPLimit$ parameter. Finally, we also show the impact of TCP flows on VoIP traffic, if dynamic voice codecs and the DCwP scheme as discussed in Chapter 5 are used and we show that this scheme can provide an increase VoIP capacity and higher TCP throughput compared to the solution based on the $TXOPLimit$ parameter alone. To the best of our knowledge, this is the first work considering TCP and dynamic VoIP flows.

2.4 Summary

The focus of this chapter was twofold. First, we provided some essential background information on voice over IP, its protocols, voice codecs as well as details about how to assess the voice call quality using the ITU E-model and the mean opinion score. This was followed by an overview of the IEEE 802.11 protocol with a focus on the Medium Access Control. Second, we reviewed research on VoIP in wireless networks, in particular focusing on the voice capacity and ways to improve the limited number of VoIP calls that can be supported.

Reviewing the literature in this area revealed that the number of VoIP calls able to be supported in an IEEE 802.11 infrastructure WLAN is surprisingly low. A number of approaches have been adopted to explore this issue, and for example it was shown that an IEEE 802.11b WLAN can only maintain 5 to 7 voice calls. It is well understood that the access point is a bottleneck in an IEEE 802.11 infrastructure WLAN and there have been multiple proposals to overcome the limited VoIP capacity. Nevertheless, proposed solutions have limitations in terms of VoIP capacity gain or required changes to hardware, software or the protocol.

In our work, we address these limitations and show how VoIP capacity in an IEEE 802.11 infrastructure WLAN can be substantially increased using existing standards without the need for additional hardware, software or changes to the protocol.

3

Analysis, Improvements and Limits of Voice over IP in IEEE 802.11 wireless LAN

In Chapter 2 we showed that the access point (AP) in an IEEE 802.11 infrastructure WLAN is a bottleneck that severely limits the number of VoIP calls that can be supported. In this chapter we study the performance of voice over IP traffic in IEEE 802.11 infrastructure WLANs, and in particular focus on the voice capacity, that is defined as the maximum number of voice calls that can be supported in a WLAN with an acceptable level of voice call quality. Our analysis will confirm that the AP is a bottleneck in a wireless LAN, thus limiting the number of calls that can be maintained. To tackle the bottleneck problem and to improve the VoIP capacity, similar to [2], we use the adjustable $TeXOPLimite$ parameter to give preference to the AP. We show that this solution can significantly (100%) increase the number of voice conversations that can be maintained with an acceptable level of quality. We show that the use of the adjustable IEEE 802.11 MAC parameter is a tradeoff between the voice capacity and the overall system performance, and we show that our findings will provide an optimal solution in terms of VoIP capacity and overall system performance along with matching our requirements for an ideal solution as outlined in Section 1.1.

This chapter is organized as follows: In Section 3.1 we give a brief problem description discussing the voice capacity limit and its cause. We then discuss our approach in Section 3.2, which is followed by the scenario description in Section 3.3, before we discuss our analytical model in Section 3.4. In Section 3.5 we discuss our validation techniques and provide an overview of the simulation environment and our IEEE 802.11 wireless testbed. In Section 3.6 we present our results and findings with regard to the overall voice capacity and the capacity gain that can be achieved by applying priority to the AP only, before we conclude this chapter in Section 3.7 with a summary of findings and contributions.

Overall, our contributions in this chapter can be summarized as follows:

1. We develop a detailed analytical model to determine the maximum number of voice calls that can be supported with an acceptable voice call quality. Using different standards (IEEE 802.11a/b/g) we confirm that the voice capacity is limited by the channel access mechanism of the IEEE 802.11 protocol rather than the available bandwidth. Note that our analytical model also accurately captures the difference between the EDCA and the DCF backoff process.

2. We show that a significant increase ($\approx 100\%$) in voice capacity can be achieved using the adjustable $TXOPLimit$ parameter of the IEEE 802.11 QoS mechanism. In particular, we show that an equal capacity gain can be achieved as in [2], but with a smaller value of $TXOPLimit$ parameter.

3. We highlight that even though the voice capacity can be increased significantly using the aforementioned $TXOPLimit$ parameter, there exists an asymptotic value for the maximum number of voice calls that can be supported when an arbitrarily large value of the $TXOPLimit$ parameter is applied, and that the VoIP capacity cannot be increased beyond that asymptotic value.

4. We find that there exists a value of the $TXOPLimit$ parameter beyond which the voice capacity does not increase. In particular we show that this value is the optimal $TXOPLimit$ value that maximizes the voice capacity without compromising the overall performance of the WLAN.

We confirm this finding by showing that a bottleneck shift will occur for value of $TXOPLimit$ larger than the optimal value, resulting in excessive packet loss and delay at the wireless nodes.

5. We show that changes to the contention window size, in particular CW_{min} in conjunction with the $TXOPLimit$ parameter has only a marginal impact on the number of voice calls that can be maintained, and that changes to the $AIFS$ parameter in a single traffic scenario will not yield a further voice capacity gain.

6. Finally, we show that our model is versatile and can also be used to accurately obtain the voice capacity in WLAN for variable bit rate traffic flows.

3.1 Overview and Problem Description

In the previous chapter we have discussed the potential of VoIP in IEEE 802.11 WLANs to replace today's mobile phone communication, as wireless VoIP can provide cost savings to the user. Even though there are a variety of challenges associated with wireless VoIP that need to be addressed, such as interference or mobility, of major concern is the limited number of VoIP calls that can be maintained with an acceptable level of quality. In Chapter 2 we highlighted that an IEEE 802.11b WLAN for example can only maintain between 5 to 7 voice calls, depending on the voice codec used. The number of calls is limited because in an infrastructure WLAN all traffic to and from the wireless nodes has to pass through the common AP, acting as a bridge between the wired and the wireless domain. Due to the design of the medium access control, each station has an equal probability of obtaining channel access. Therefore, with an increasing number of wireless nodes in the WLAN, the probability of the AP winning the channel contention is reduced, because the AP has to compete against all wireless nodes to access the channel for each packet on the downlink.

Even though 5 to 7 concurrent voice calls may be sufficient in a home network environment, it is insufficient in office environments, where there are commonly many users connected to the WLAN, or in public settings, such as in airport lounges or restaurants. Furthermore, the 5 to 7 voice calls is far below the expected number of calls if the number of calls is determined based on a simple bit rate calculation as shown in Section 2.3. Given that the limited VoIP capacity potentially slows the adoption of wireless VoIP, solutions need to be developed to overcome this limitation.

3.2 Improving the VoIP capacity

In Section 2.3 we discussed a variety of solutions that have been proposed in the literature to address the limited VoIP capacity. However, as we have shown, none of the solutions match our requirements for an optimal solution defined in Chapter 1, or impact on the overall system performance.

Our approach is based on an analytical model, and we use simulation and real testbed measurements to validate our findings and results. Similar to the work in [102, 103], we develop an analytical model using a mean-based approach, whereby we calculate, for example, the average service time of a packet, that can then be used to derive the number of VoIP calls that can be supported. Whereas the work in [102, 103] assumes that a station has an infinite buffer, our approach is more realistic as we consider a finite buffer at a station. In particular, we model a station based on an $M/G/1/K\epsilon$queueing system where the station can queue up to $K\epsilon$packets. Note that in Chapter 4 we will relax the buffer restriction to some degree and show that a similar number of voice conversations can be supported. Using an analytical model allows us to gain a better understanding of the interactions between the VoIP packets and the contention based channel access. This insight into these interactions then allows us to develop a solution to the VoIP capacity problem.

We confirm that an $M/G/1/K\epsilon$queueing system can be used to model a station, by comparing the analytical results with those obtained by simulation and testbed measurements. We show that the results are in good agreement and are in line with those reported in the relevant literature, discussed in Chapter 2.

Further to our voice capacity analysis, we show that a significant voice capacity gain can be achieved using the adjustable MAC parameter to give priority to the AP. Even though some of the proposed solutions to increase the number of voice calls in a WLAN show significant improvements, most of them require additional hardware or modification to the protocol. However, due to the required changes, a large scale adoption is unlikely, in particular for solutions that require changes to the protocol, different channel access controls or changes to the call admission control [127, 139, 143, 171, 172]. In our approach we aim for a solution that boosts the voice capacity and that also can be readily implemented using existing hardware and software and does not require any changes to the protocol. Even though the work in [2, 130, 145] uses a similar approach, we show that the proposed parameter settings are not ideal and impact on the overall performance of the WLAN. For example, we will show, that setting a value of the $TXOPLimit$ parameter that is too large will negatively affect the overall performance in the WLAN to an extent that the call quality is no longer acceptable. In particular we show that assumptions made in [2] will not hold in general and that we can achieve the same capacity increase with a smaller value of the $TXOPLimit$ parameter. Specifically, we show that our MAC parameter settings are optimal in terms of voice capacity and overall system performance.

3.3 Scenario and system configuration

In this scenario we consider an IEEE 802.11e infrastructure WLAN as shown in Fig. 3.1.

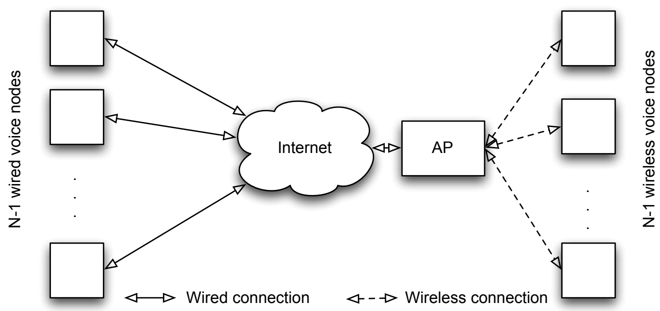

Figure 3.1: Infrastructure VoIP WLAN topology

In this network voice traffic to and from any mobile node must flow through the common AP acting as a base station. Since every station including the AP has the same chance to access the wireless channel, the probability of the AP winning a channel access decreases with an increasing number of wireless nodes that maintain a voice call. This is because the AP has to compete against all wireless nodes to access the channel for every packet of the downlink stream. As the probability of the AP winning channel access decreases, the AP becomes a bottleneck and packets from the downlink streams start to build up at the AP buffer. If the number of voice calls keeps increasing, at some stage, the AP will start to drop packets due to buffer overflows and the quality of the voice call starts to degrade. A variety of network metrics can be used to determine the user perceived voice call quality. Here we use packet loss as an indicator and define a packet loss threshold, over which the satisfactory user-perceived quality for a call cannot be maintained.

In our analysis we use the network, MAC and traffic parameters as outlined in Table 3.1.

Network & IEEE 802.11 MAC parameter			
PHY	802.11b	802.11a	802.11g
Channel rate	11MBit/s	54Mbit/s	54MBit/s
Basic rate	1MBit/s	1MBit/s	1MBit/s
Backup Slot length $\sigma\epsilon$	20 μs	9 μs	9 μs
CW_{min}	31	15	15
CW_{max}	1023	1023	1023
T_{SIFS}	10 μs	16 μs	10 μs
T_{AIFS}	50 μs	34 μs	28 μs
Max. backoff stage (m)	5		
Retry limit (R)	7		
Buffer size (K)	50 packets		
Traffic/Data details			
PLCP & Preamble	192 μs	20 μs	20 μs
MAC Header + PCS	24.7 μs	5 μs	5 μs
RTP/UDP/IP Header	29.1 μs	5.9 μs	5.9 μs
Voice payload	$\frac{L\times 8}{\text{Channel rate}}\mu s\epsilon$		
ACK frame	112 μs		

Table 3.1: Network parameters for IEEE 802.11a/b/g wireless LAN

3.4 The analytical model

Let us consider an IEEE 802.11 infrastructure WLAN consisting of one access point (AP) and $N\epsilon-\not{1}$ wireless nodes. Each wireless node maintains a full-duplex VoIP call to a node outside of the wireless network using the same voice codec. Here we only consider voice traffic and no other cross traffic, such as TCP traffic. We assume EDCA basic access is used over an ideal channel without interference or hidden terminals. Unless stated otherwise, the indices $a\epsilon$ and $n\epsilon$correspond to the access point and a wireless node, respectively.

Let λ_n be the packet arrival rate of a wireless node in the network shown in Fig. 3.1. The arrival rate at the AP is a superposition of all the individual rates of voice traffic from $N\epsilon-\not{1}$ wireless nodes and is given by $\lambda_a = (N\epsilon-\not{1})\lambda_n$.

Denote the packet service rate of the AP and a wireless node by μ_a and μ_n, respectively. Assuming packets arrive at a node according to a Poisson process, an $M/G/1/K$ queueing model can then be used to model the wireless node and the AP, where the latter serves packets in batches of η packets. Note that K is the number of packets that can be queued at a *station* (wireless node and/or AP) and can take different values at the wireless node and the AP. Note that even though the assumption of packets arriving as a Poisson process is coarse, as the packet arrival of an individual call is deterministic [173], it can, in part, model the superposition of multiple periodic streams at the AP, which in turn is the bottleneck determining the VoIP capacity in this network. Using an $M/G/1/K$ model enables us to study the relation between the average arrival and services rates which are the key to determining the number of voice calls that can be supported. In [174] it has been shown that the Poisson assumption is an adequate approximation. To this end we have conducted extensive simulation and measurements over an IEEE 802.11 WLAN testbed (see Section 3.6) with a range of parameters using periodic streams and confirm the validity of the Poisson approximation.

The queue utilization can be expressed as $\rho_i = \lambda_i/\mu_i,\ i \in \{n, a\}$ where ρ_i is also the probability that a station has a packet to send. Thus a station will be idle with probability $1 - \rho_i$. Our analysis is based on a fixed-point formulation between the collision probability seen by a packet transmitted on the channel and the attempt rate per slot of a station. The latter is conditioned on a station that has packets to send, i.e. the queue is nonempty. In the following we establish the fixed-point equations and derive the average service times associated with the AP and wireless node, respectively.

As outlined in Chapter 2 the IEEE 802.11 protocol [4] specifies that a station has to wait a random period of time measured in backoff slots before attempting to transmit its packet. The backoff is uniformly and randomly selected from $[0, CW - 1]$, where CW is the current contention window with the initial value of $CW_{min} = W$. Collision occurs if more than one station transmits in the same slot. If a collision occurs, the contention window of the sender is doubled (unless the maximum value ($CW_{max} = 2^m CW_{min},\ m \geq 1$) has been reached) and the packet is scheduled for retransmission. The contention window is reset to CW_{min} when the packet either has been successfully transmitted or discarded when the retransmission limit R ($R \geq m$) is reached.

Let $\tau_i,\ i \in \{n, a\}$ be a conditional attempt rate per slot of a station (i.e. the ratio of the number of attempts to the time spent in backoff measured in slots provided that the station has packets to send). Knowing that ρ_i is the probability that a station has packets to send, the probability that a station is attempting to transmit in a slot is given by $\rho_i \tau_i,\ i \in \{n, a\}$ [175]. Assuming each packet collides with constant and independent probability $c_i,\ i \in \{n, a\}$, a fixed point formulation can be established for a wireless node and the AP, respectively as

$$
\begin{aligned}
c_n &= 1 - (1 - \rho_n \tau_n)^{N-2}(1 - \rho_a \tau_a), && (3.1)\\
c_a &= 1 - (1 - \rho_n \tau_n)^{N-1}, && (3.2)
\end{aligned}
$$

where $\rho_n \tau_n$ and $\rho_a \tau_a$ is a function of c_n and c_a. Note that Eq. (3.1) and Eq. (3.2) are based on the fact that transmission from a wireless node can collide with a transmission from one of the remaining $N-2$ wireless nodes or the AP, whereas the AP can only collide with one of the $N-1$ wireless nodes. To complete the fixed point equations, we first devise the average service time of a station and then determine the function $\rho_i \tau_i,\ i \in \{n, a\}$.

As in [103], the average packet service time of a wireless node can be decomposed into three components: i) the actual collision and successful transmission time of the packet; ii) the interruptions to the backoff due to collisions and successful transmissions by the remaining $N-2$ wireless nodes and the AP; and iii) the average backoff during which the channel is sensed idle.

For the first component, the collision and successful transmission time of a voice packet is defined as

$$
\begin{aligned}
T_c &= T_p + T_{ACK_{TO}} + T_{AIFS}, && (3.3)\\
T_s &= T_{AIFS} + T_p + T_{SIFS} + T_{ACK}, && (3.4)
\end{aligned}
$$

where T_p, T_{ACK} are the transmission times of the packet and acknowledgement, $T_{ACK_{TO}}$ is the ACK timeout period of an unsuccessful transmission, and T_{AIFS}, T_{SIFS} are the duration of the arbitrary and short inter-frame spaces in $[\mu s]$, respectively. As a packet can collide several times before its successful reception, the average collision time $\bar{t}_n$ caused by a single wireless node during a packet service time $1/\mu_n$ is given by

$$
\frac{\bar{t}_n}{2} = \frac{1}{2} \sum_{i=1}^{R} c_n^i (1 - c_n) i T_c \underset{R \to \infty}{\approx} \frac{T_c\, c_n}{2(1 - c_n)}. \qquad (3.5)
$$

Note that the $1/2$ factor in the above expression is based on the assumption that a collision is due to simultaneous transmissions from two stations only, and thus the average collision time caused by a wireless node is half of the total collision time experienced by all stations. Similarly, the average collision time caused by the AP is given by

$$\frac{\bar{t}_a}{2} = \frac{T_c\, c_a}{2(1-c_a)}, \tag{3.6}$$

where T_c is defined as in Eq. (3.3) because only the first packet in the transmitted TXOP-frame from the AP would experience a collision.

For the second component, the average packet service time of a wireless node $1/\mu_n$ consists of the average successful transmission time given as

$$(N-2)\lambda_n \frac{1}{\mu_n} T_s + \frac{\lambda_a}{\eta}\frac{1}{\mu_n}(T_s + (\eta-1)T_s^*) \tag{3.7}$$

and the average collision time of the other $(N-2)$ wireless nodes and the AP

$$(N-2)\lambda_n \frac{1}{\mu_n}\frac{\bar{t}_n}{2} + \frac{\lambda_a}{\eta}\frac{1}{\mu_n}\frac{\bar{t}_a}{2}. \tag{3.8}$$

Note that $T_s^* = T_p + 2T_{SIFS} + T_{ACK}$ and $\bar{t}_a/2$ in Eq. (3.8) is defined in Eq. (3.6). Also note that the term λ_a/η in Eq. (3.7) and Eq. (3.8) is due to the fact that for each channel access the AP can send up to η packets in its TXOP-frame.

For the third component, the average backoff duration of a wireless node $\overline{w}_n$, each transmission attempt has a collision probability c_n and a successful transmission probability of $(1-c_n)$. Recall that for each collision the contention window is doubled unless the maximum contention window size has been reached, i.e. $W = CW_{max} = 2^m CW_{min}$. Then with a maximum backoff stage m and a retry limit R, the average backoff can be calculated using

$$\begin{aligned}
\overline{w}_n &= (1-c_n)\frac{W-1}{2} + (1-c_n)c_n\frac{2W-1}{2} \\
&\quad +(1-c_n)c_n^2\frac{2^2W-1}{2} + \ldots + c_n^{R-1}\frac{2^mW-1}{2} \\
&= \sum_{i=0}^{R-2}(1-c_n)c_n^i\frac{\left(\prod_{j=1}^{i}\alpha_j W\right)-1}{2} + c_n^{R-1}\frac{2^mW-1}{2},
\end{aligned} \tag{3.9}$$

where $\alpha_j = 2$ if $j \leq m$, and 1 otherwise. Note that $\overline{w}_n$ in Eq. (3.9) is given in backoff slots, each of a length of $\sigma[\mu s]$.

Based on Eq. (3.9) the conditional attempt rate per slot of a wireless node can be immediately derived as

$$\tau_n = \frac{\sum_{i=0}^{R} c_n^i}{\overline{w}_n}, \tag{3.10}$$

where the numerator is the average number of transmission attempts per packet of the wireless node during its backoff period.

However, in contrast to the model in [103], the actual length in μs of the average backoff is not simply $\overline{w}_n \sigma$ because the backoff counter is managed differently in EDCA [89]. In particular, after every channel activity, the backoff counter in EDCA is resumed one slot time before the $AIFS$ timer elapses. On average, during the backoff of a wireless node, the number of times the channel is sensed busy is given by

$$ {}_n = 2\rho_n \left((N-2) + \frac{N-1}{\eta} \right), \tag{3.11}$$

which can be derived based on the average number of successful transmissions and collisions from Eqs. (3.7) and (3.8) and the fact that arrival of packets at the AP is the superposition of many individual voice traffics in that network, i.e.,

$$\lambda_a = (N-1)\lambda_n. \tag{3.12}$$

Furthermore, according to the EDCA mechanism, a station has to wait either an additional slot when the medium is idle or an additional $AIFS$ period when the medium is sensed busy, before attempting to transmit. The former occurs with a probability $1 - c_n$, while the latter with a probability c_n. Thus the third component in $[\mu s]$ can be expressed as

$$(\overline{w}_n - /_n)\sigma + (1 - c_n)\sigma + c_n T_{AIFS}. \tag{3.13}$$

Based on Eqs. (3.4), (3.5), (3.7), (3.8) and (3.13) the average packet service time of a wireless node is given by

$$\begin{aligned} \frac{1}{\mu_n} &= ((N-2)\rho_n + 1) \left(T_s + \frac{\bar{t}_n}{2} \right) \\ &+ \frac{N-1}{\eta}\rho_n \left(T_s + (\eta-1)T_s^* + \frac{\bar{t}_a}{2} \right) \\ &+ (\overline{w}_n - /_n + 1 - c_n)\sigma + c_n T_{AIFS}. \end{aligned} \tag{3.14}$$

We now find the attempt probability and average service time of the AP. By replacing c_n with c_a in Eq. (3.10) the conditional attempt rate per slot of the AP immediately follows as

$$\tau_a = \frac{\sum_{i=0}^{R} c_a^i}{\overline{w}_a}, \qquad (3.15)$$

where $\overline{w}_a$ is the average backoff of the AP given by

$$\overline{w}_a = \sum_{i=0}^{R-2} (1 - c_a) c_a^i \frac{\prod_{j=1}^{i} \alpha_j W - 1}{2} + c_a^{R-1} \frac{2^m W - 1}{2}. \qquad (3.16)$$

Because the AP is allowed to send up to $\eta \geq 1$ packets consecutively, the average service time of the TXOP-frame consists of two parts: (i) the average service time of the first packet in the frame ($1/\mu_{a1}$), and (ii) the total average service time of all the subsequent packets in that frame ($1/\mu_{a2}$).

The average service time of the first packet in a TXOP-frame is calculated in a similar way to Eq. (3.14) by appropriately using the collision probability c_a instead of c_n. Therefore the first part of the average service time of the TXOP-frame is given by

$$\begin{aligned} \frac{1}{\mu_{a1}} &= (N-1)\frac{\lambda_n}{\mu_a}\left(T_s + \frac{\bar{t}_n}{2}\right) \\ &+ \left(T_s + \frac{\bar{t}_a}{2}\right) \\ &+ (\overline{w}_a - {}_a + 1 - c_a)\sigma + c_a T_{AIFS}, \end{aligned} \qquad (3.17)$$

where $\bar{t}_a/2$ is given in Eq. (3.6), $\overline{w}_a$ in Eq. (3.16) and ${}_a$ is defined as

$${}_a = 2(N-1)\frac{\lambda_n}{\mu_a}. \qquad (3.18)$$

After the first packet of the TXOP-frame is served, all the subsequent packets in that frame have an equal service time of T_s^* $[\mu s]$. This is because the remainder of packets in the frame do not need to contend for the channel access. Thus

$$\frac{1}{\mu_{a2}} = (\eta - 1)T_s^*, \qquad (3.19)$$

and the average service time of a *packet* sent by the AP can then be calculated as follows:

$$\frac{1}{\mu_a} = \frac{1}{\eta}\left(\frac{1}{\mu_{a1}} + \frac{1}{\mu_{a2}}\right). \qquad (3.20)$$

Equations (3.1), (3.2), (3.10), (3.14), (3.15) and (3.20) constitute a set of fixed-point equations that can be solved iteratively to obtain the collision probabilities c_i and the conditional attempt probability τ_i, as well as $\rho_i = \lambda_i/\mu_i,\ i \in \{n, a\}$.

Having obtained $\rho_i,\ i \in \{n, a\}$, we require the average packet loss κ of a voice call to be less than 2% to have an acceptable level of quality [65]. The maximum number of supported voice calls (C) is the number of calls such that the packet loss probability at the AP (p_a) of a voice call is kept to less than κ. Because the average packet loss seen by the AP is also the average packet loss of an individual call, and the buffer at the AP is the bottleneck, p_a can then be approximated by

$$p_a \approx \frac{(1-\rho_a)\rho_a^K}{1-\rho_a^{K+1}}. \tag{3.21}$$

To obtain C, we repeatedly solve the above non-linear system of equations by incrementing the number of voice calls. Although the voice quality depends on both the average packet loss and the end-to-end delay, capacity C is calculated based on the packet loss only. This is because when the AP is the bottleneck, packet loss will be the main factor causing degradation in voice quality. We will look at the delay as well when this bottleneck is shifted as shown in the next section. Note that Eq. (3.21) is a blocking probability of an $M/M/1/K$ queue [176] by assuming exponential service time at the AP and thus it is only an approximation for the packet loss probability. As the $M/G/1/K$ model is a generalized $M/M/1/K$ model where the service time μ is no longer exponential [118], we can use the blocking probability in Eq. (3.21) as an approximation for the packet loss probability of the $M/G/1/K$ model. In the next section, we show that this approximation is reasonable and matches well with the packet loss observed from simulations and testbed measurements. Also we assume here that packet loss is only due to the buffer overflow at the AP, and not caused by packet collision or channel errors.

3.5 Validation

3.5.1 Simulation

To validate the results obtained by the analytical model, we performed simulations using the common ns-2 (version 2.28) [177] simulation tool. To enable quality of service, we use the EDCA extension from the TU-Berlin [178], that implements the four different access categories and allows the adjustment of different MAC parameters. More details about the EDCA extension are available in [179, 180].

The simulation WLAN consists of a single wired node, an access point (AP) and $N - 1$ wireless voice nodes (WNs), as depicted in Fig. 3.2.

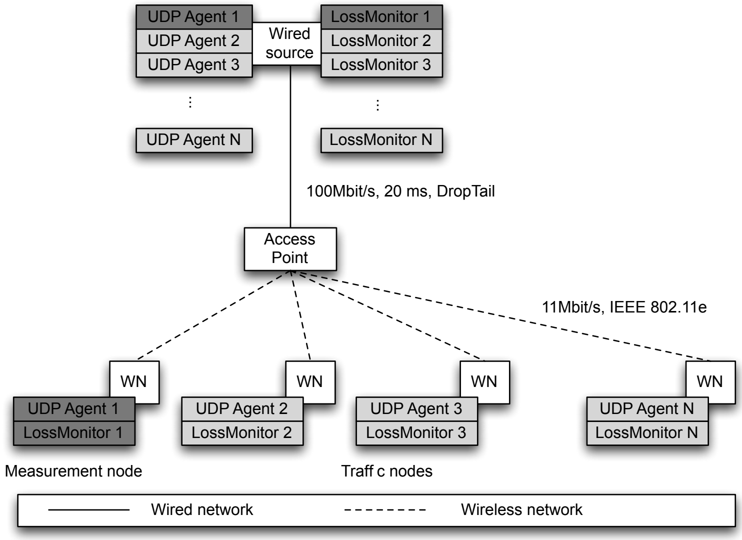

Figure 3.2: NS-2 simulation IEEE 802.11(e) testbed

In the simulation we consider multiple full-duplex voice calls between the wired source outside the WLAN and the different wireless voice nodes. In this setup, the wired node is used to generate the downlink voice flows and to receive the uplink voice flows of the wireless voice nodes and, as shown, is directly connected to the AP. The AP is configured to uses an IEEE 802.11b WLAN with EDCA enabled. Here each voice station (wired/wireless) uses a UDP agent (`[Agent/UDP]`) that generates a constant bit rate traffic flow with an average packet rate of λ packets per second (`[Application/Traffic/CBR]`) to emulate voice flows with different voice codecs, such as a G.729 or G.711 voice codec with different sampling rates. All parameter settings, for example CW_{min} or the buffer size K, are set according to the values shown in Table 3.1 in Section 3.3.

A loss monitor (`[Agent/LossMonitor]`) is used to obtain different traffic metrics, such as received bytes and the number of packets received. The simulation uses a *recording* function to read the different metrics and to write (record) these details into a log-file on a per second base. During the post-processing a variety of statistics, such as packet loss and delay are derived based on this output as well as additional log-files.

Each simulation was run such that every 10 seconds and additional voice call was added to the wireless network. For example, at time T_0[1] the simulation starts; at T_{10} the first VoIP call is added to the WLAN, at T_{20} the second call is added and so forth, until the maximum number of calls is reached indicated by packet loss above the QoS threshold of 2%. The average packet loss is then calculated over a 10 s time period using $100 - \cancel{(}P_{RX}/P_{TX}) \times \cancel{1}00$, where P_{RX} is the total number of packets received and P_{TX} is the total number of packets sent. To assure statistical accuracy, the average packet loss is also reported using a 95% confidence interval. Note that because of the difference in obtaining packet loss between the simulation and the analytical model and because the analytical model may not capture all interactions observed in the simulation, there may be a slight difference in results obtained using the model and the simulation. Nevertheless, we will show in later sections and other chapters, that overall the simulation results are in good agreement with results obtained analytically.

Finally, to differentiate between traffic from the AP and the wireless nodes, the AP and the wireless nodes were placed in different access categories where all MAC parameters were set equal. This is achieved by setting a different value for the `prio_` parameter in the simulation script, whereby the different parameter for each access category are set in the `priority.tcl` file located in a `mac/80211_e/` subdirectory as outlined in [179,180] and shown in Appendix A.

3.5.2 WLAN testbed

In addition to simulation, we used an IEEE 802.11 infrastructure WLAN testbed to further validate our results obtained using the analytical model. The wireless testbed consisted of a single AP and multiple wireless nodes. In this testbed, a desktop PC is used as an AP and the wireless nodes are a combination of desktop PCs, Netbooks as well as embedded devices, and is shown in Fig. 3.3, and an overview of the system hardware and operating system is given in Table 3.2.

Each system is equipped with an Atheros[2] 802.11 wireless card using a version of the MADWIFI[3] driver. The MADWIFI wireless drivers are used because of their support of the IEEE 802.11 QoS mechanism. In particular, the

[1] T_ii = time in seconds
[2] http://www.qca.qualcomm.com/
[3] http://www.madwifi-project.org

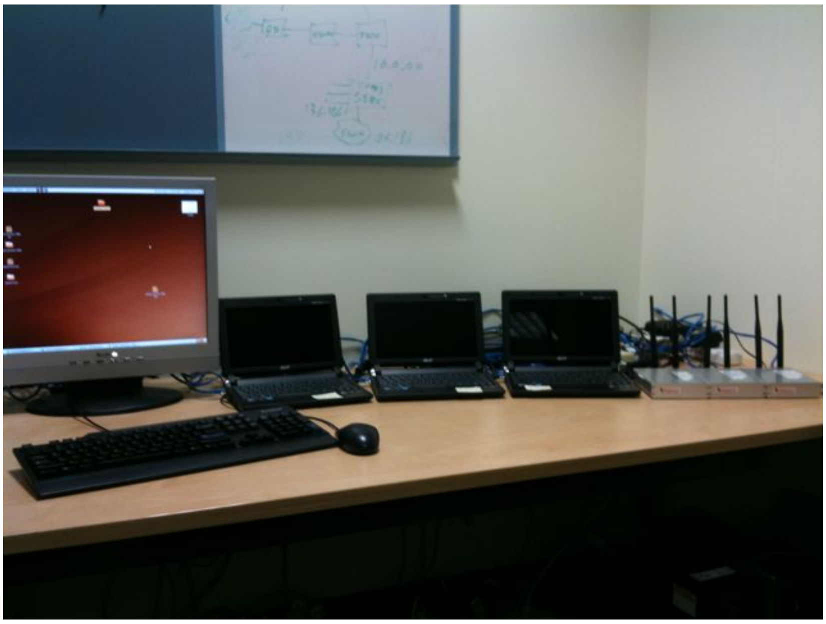

Figure 3.3: IEEE 802.11 wireless testbed at the Centre for Advanced Internet Architectures (CAIA)

IEEE 802.11 QoS mechanism is supported using WME (Wireless Multimedia extensions) [181]. To be backwards compatible, WME can be deactivated if required. In this case, the MADWIFI driver use the default IEEE 802.11 DCF instead of EDCA. Enabling WME also allows the adjustment of the $AIFS$, $TXOPLimit$ and the contention window MAC parameter.

In Figs. 3.4 and 3.5 we show the default WME parameters for the Linux and the embedded devices. As shown, the wireless multimedia extension supports eight different user priorities with four access categories, best effort (AC_BE), background (AC_BK), video (AC_VI) and voice (AC_VO) and allows the adjustment of CW_{min}, CW_{max}, $AIFS$ and the $TXOPLimit$. Note that there is a difference in setting the $TXOPLimit$ between Linux and and embedded device running TinyBSD. As shown, the Linux device specifies the $TXOPLimit$ in milliseconds [ms], whereas in TinyBSD the $TXOPLimit$ is set as an integer in units of 32 μsec (e.g. $94 \times 32 \mu sec = 3008 ms$). Also note that the shown $TXOPLimit$ parameter is set for an IEEE 802.11a/g WLAN and requires adjustment for an IEEE 802.11b WLAN. The $AIFS$ and contention window size values are set in slots and the slot length depends on the underlying PHY and in this thesis are set according to the parameters specified in Table 3.1.

The embedded Unix devices are equipped with an AMD Geode CPU, 128 MB Ram and use a 512 MB compact flash card as main storage. As shown in Fig. 3.6 the embedded device has two wireless interfaces and a single wired

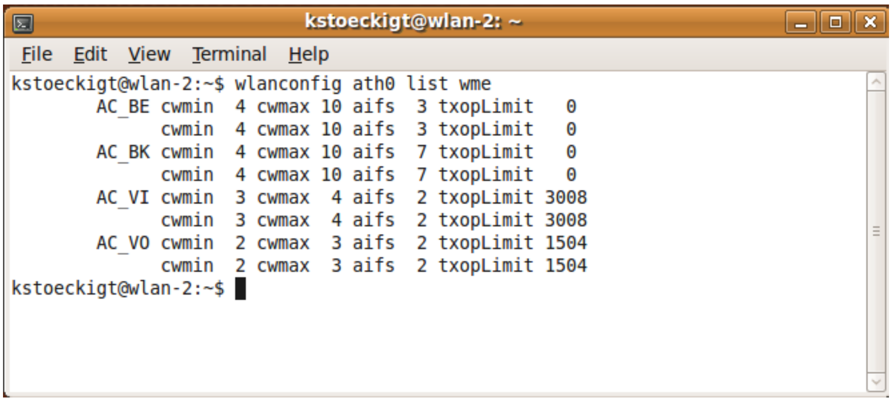

Figure 3.4: WME parameter on Linux devices

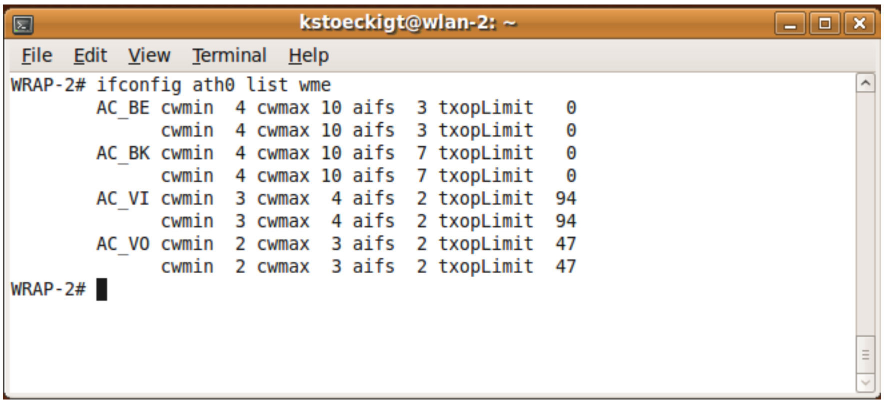

Figure 3.5: WME parameter on embedded devices (TinyBSD)

System	OS	WLAN	Chipset
3 x Desktop PC	Linux 2.6.32	802.11a/b/g	AR5413
5 x Acer Netbooks	Linux 2.6.32	802.11b/g	AR928X
3 x Embedded devices	TinyBSD 6.2-Release	802.11a/b/g	AR5212

Table 3.2: System overview

ethernet port. Whereas the desktop PCs and the Netbooks run a Linux 2.6.32 as the operating system, the embedded devices are configured with TinyBSD 6.2 [182].

Figure 3.6: Embedded Linux device

The network is setup such that the wireless devices connected to the common AP using a private subnet. Furthermore, each of these devices was also connect to a wired control network, as shown in Fig. 3.7, that was used to send different commands to the wireless nodes without interfering with the WLAN. Also, for the measurements, the RTS/CTS mechanism has been deactivated along with other manufacturers' features such as *fast-frames* or *bursting* and the channel rate was fixed to 11MBit/s on a pre-selected channel using the IEEE 802.11b PHY. The WLAN testbed is located in an office at the Centre for Advanced Internet Architectures[4], and to limit potential interference by other wireless devices operating in the 2.4GHz range in the vicinity of the testbed, measurements were performed during the late afternoon, evenings or on weekends.

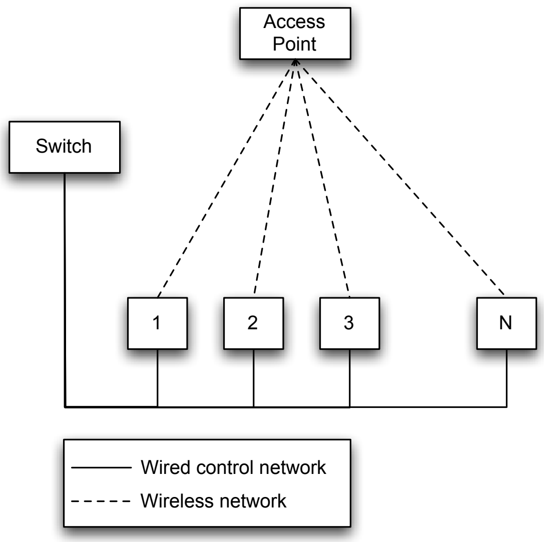

Figure 3.7: Stylized wireless testbed at CAIA

The wireless VoIP traffic was generated using the MGEN[5] traffic generator (version 4.2) with a modified log-file generator. The measurement data obtained in the testbed included packet loss, delay, throughput and jitter. The MGEN scripts were setup such that the traffic pattern would correspond to G.711 voice calls with a 10 ms sampling rate. Henceforth, MGEN generated 100 packets per second per uni-directional call (200 packets per bi-directional call) and each packet had a payload size of 80 bytes. Note that the minimum payload size MGEN can support is 28 bytes. The minimum payload size however, is larger than the payload of of G.729 voice packets that have been generated with either a 10 ms or 20 ms sampling rate. However, the slightly increased payload increases the transmission time of the payload only from 7.27 μsec to 20.36 μs, and the difference of 13.09 μsec on the total successful transmission time of a packet is negligible, and thus will not affect the results in terms of the number of calls that can be supported.

[4]CAIA, http://caia.swin.edu.au
[5]http://cs.itd.nrl.navy.mil/work/mgen/

The measurements were setup such that a new bi-directional call was added to the WLAN every 10 seconds. Due to the limited equipment, a maximum of 10 voice calls can be simulated. The measurement results were processed with a variety of scripts to extract the required information. The modified log-files were used and processed with the TRPR[6] tool that generated a variety of plots, e.g. the interarrival time (jitter) at a node.

Our IEEE 802.11 WLAN is somewhat similar to testbeds described in the literature [2, 27, 64, 65, 108, 111, 145]. Whereas [64, 108] use an in-house voice traffic generator (VGEN) with a similar WLAN setup to the CAIA WLAN, [27, 109] use the ORBIT[7] WLAN testbed for their measurements. The WLAN testbed in [2] is similar to our WLAN testbed, because it consists of a variety of embedded devices, desktop and laptop computers, and also uses the MADWIFI drivers. Similar to [2] we also adjusted the buffer settings in the network driver.

3.6 Results and Discussion

3.6.1 DCF = EDCA

As briefly outlined in Chapter 2 and Section 3.4 there exists a difference between the backoff behavior of DCF and EDCA, which affects the voice performance in the WLAN. The IEEE 802.11 protocol defines for DCF that the backoff counter is resumed after the AIFS timer has elapsed and that a station can attempt to transmit to the channel whenever its backoff counter reaches zero. In contrast, in EDCA, the backoff counter is resumed one slot time before the AIFS timer elapses. Thus, whenever the backoff counter is interrupted due to a busy channel, an EDCA station gains one time slot compared to a DCF station. Additionally, an EDCA station has to wait either an additional slot or an AIFS period if its counter has reached zero, before attempting to access the channel, depending on whether the medium has been sensed free or busy.

In Fig. 3.8 we show the average packet service rate of the AP for G.729 voice calls with a 10 ms sampling rate using an EDCA and DCF backoff process with a default TXOP value, as well as the packet arrival rate at the AP. Assuming $T_{AIFS} = T_{DIFS} = 50\mu s$, then the AP achieves a slightly higher average packet service rate when EDCA is used. Note that in Fig. 3.8 we show trendlines demonstrating the decreasing average packet service rate with an increase in packet arrival rate for an increasing number of VoIP calls, rather than

[6] http://pf.itd.nrl.navy.mil/proteantools/trpr.html
[7] http://www.orbit-lab.org/

discrete points. Results obtained in this chapter are based on this increased service rate using the EDCA mechanism. Even though the difference in service rate has no impact on the overall VoIP capacity, it is important from the modeling point of view, as other internal parameters such as the collision probability are affected by it.

Unless otherwise stated, in this thesis we consider EDCA and the difference in backoff behavior is captured by our analytical model. A detailed analysis and discussion about the impact of the different backoff behaviors of DCF and EDCA can be found in [89].

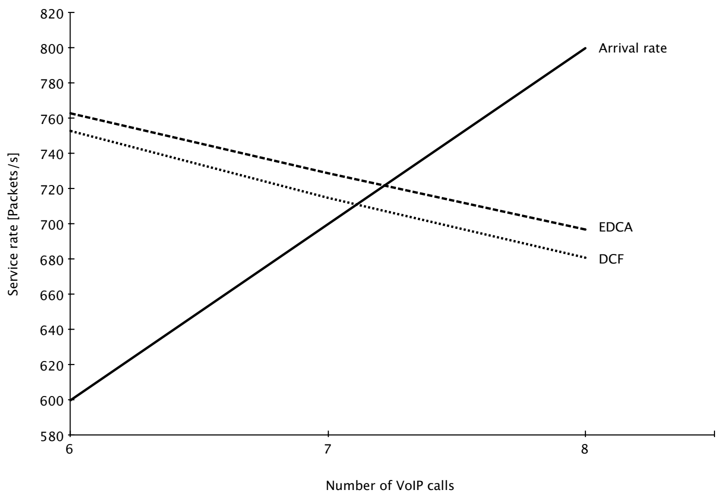

Figure 3.8: Average packet service rate at the AP using the DCF and EDCA backoff for a G.729 voice call with a 10 ms sampling rate and default TXOP parameter ($\eta = 1$), obtained analytically

3.6.2 VoIP capacity

Before we evaluate the performance gain that can be achieved using the adjustable $TXOPLimit$ parameter, we confirm the limited voice capacity in IEEE 802.11 WLAN using our analytical model where the $TXOPLimit$ parameter η is set to 1.

In Fig. 3.9 we show the average number of packets queued at the AP and the wireless node with an increasing number of voice calls in the WLAN. As shown, with an increasing number of voice calls in the WLAN, packets are queued at the AP and the wireless nodes. As the maximum buffer size is set to $K = 50$ packets, once the queue at a station is saturated, packet loss occurs. As shown, the WLAN can maintain up to 6 voice calls using a G.711 voice codec with a 10 ms sampling rate before the queue size would exceed its limit. For

G.729 voice calls, the AP buffer size is exceeded once the 8th call joins the WLAN, hence only 7 calls can be supported. Similar results can be observed for the voice codecs when a 20 ms sampling rate is used. It can also be seen that the number of queued packets at the wireless nodes also increases. However, as on average a wireless voice node can access the channel more frequently, and also has a lower packet arrival rate than the AP, only a small number of packets are queued at any given time. Observe that the queue utilization at a wireless node is marginally higher when a G.711 voice codec is used. This is because the payload size of a G.711 packet is 80 bytes, and thus the total transmission time is larger than that of a G.729 voice packet with a 10 byte payload. Furthermore, it can be observed that the average number of queued packets at a wireless station is approximately halved if a G.7xx voice codec with a 20 ms sampling rate is used. This is because the voice codec only generates half as many packets per time unit, thus, on average halving the number of queued packets.

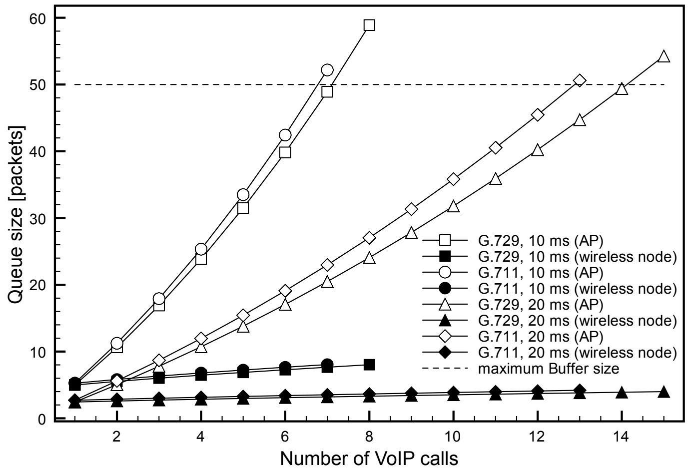

Figure 3.9: Average number of packets queued at the AP and the wireless node for G.729 and G.711 voice calls with a 10 ms and 20 ms sampling rate, obtained analytically

Our analytical results confirm the limited VoIP capacity as reported in the literature. Note that here $\eta = 1$, that is, a station can transmit a single packet per channel access. We will demonstrate the benefits of the previously discussed solution using $\eta > 1$ in the next section.

3.6.3 VoIP capacity improvements using $TXOPLimit$

In this section we show the capacity gain that can be achieved when preference is given to the AP using the adjustable $TXOPLimit$ parameter. Recall that we define the $T\epsilon XOP\epsilon Limit\epsilon$ parameter as packets per channel access and denote the parameter as η.

In Fig. 3.10 we show the packet loss probability at the AP obtained using Eq. (3.21) for G.729 and G.711 voice calls with 10 ms sampling rate and selected values of TXOP parameter η. It can be observed that for $\eta\epsilon = 1$ (default TXOP) the network can accommodate up to 7 voice calls using a G.729 codec before the packet loss threshold $\kappa\epsilon = 0.02$ (2%) is exceeded. For $\eta\epsilon = 5$ the voice capacity is almost doubled with 12 voice calls using the same codec. A similar number of voice calls is reported in [2], but with a larger TXOP value where the authors set the TXOP value equal to the number of expected voice calls. Also observe that the measurement results obtained using our testbed for a G.711 voice call with a 10 ms sampling rate match the analytical results closely. Note that due to limited testbed equipment, experimental results are only shown for the TXOP parameter $\eta\epsilon = 1$ and $\eta\epsilon = 2$. Also note that items marked by an asterisk (*) are results obtained through measurements in a WLAN testbed, and to mitigate for fluctuations in the obtained results, caused by potential interference, the WLAN measurement results are shown with a 95% confidence interval

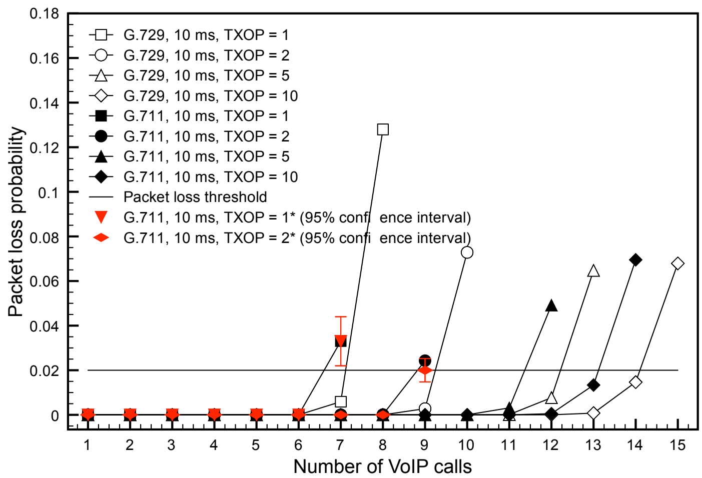

Figure 3.10: Packet loss probability at the AP for G.729 and G.711 voice calls with a 10 ms sampling rate and selected values of TXOP, obtained analytically and by testbed measurements (indicated using an asterisk (*)).

In Table 3.3 we show analytical results for the number of calls an IEEE 802.11b WLAN can support for G.729 and G.711 voice codecs with a 10 ms and 20 ms sampling rate and selected values of TXOP parameter η.

The increase in VoIP capacity with increasing values of $T\epsilon XOP\epsilon Limit\epsilon$ parameter is because the average queue utilization at the AP is significantly lower,

$\eta\epsilon$	Voice codec G.729, 10 ms	G.711, 10 ms	G.729, 20 ms	G.711, 20 ms
1	7	6	14	12
5	12	11	24	21
10	14	13	27	24

Table 3.3: Number of voice calls in an IEEE 802.11b WLAN with an 11 MBit/s data rate and selected values of TXOP parameter, obtained analytically

as the AP can now transmit multiple packets per channel access without contending for the channel again. The reduction in AP queue utilization, redi.e. a fewer number of packets queued at the AP is shown in Fig. 3.11 for the G.729 and G.711 voice codec with selected values of η, obtained using the analytical model.

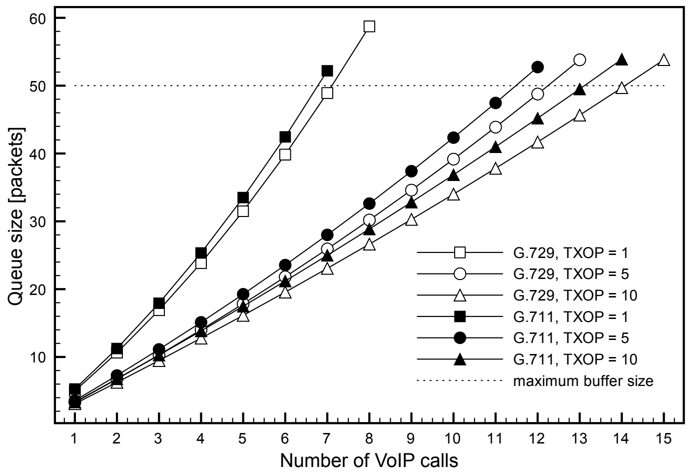

Figure 3.11: Number of packets queued at the AP for an increasing number of G.729 and G.711 VoIP calls with selected values of $TXOPLimit$ parameter η, obtained analytically

In Figs. 3.12 and 3.13 we compare our analytical results with simulation using G.729 and G.711 codecs, respectively. Observe that the analytical results match those of the simulation closely. Note that in some cases there is a one call difference between the analytical and simulation results. This is because the analytical model uses the approximation equation Eq. (3.21) to obtain the packet loss, whereas in the simulation, packet loss is obtained using $100-(\text{total number of packets received}/\text{total number of packet sent})\times 100$. The difference in obtaining packet loss along with the approximation equation in the analytical model can potentially induce some inaccuracies that cause the slight difference. This means for example that the packet loss obtained using Eq. (3.21) shows loss in excess of the acceptable threshold, whereas the results obtained by simulation are still below the packet loss threshold of 2%, and vice versa, thus leading to the one call difference in some cases. Overall, however, the analytical and simulation results are in good agreement for a wide range of parameter settings.

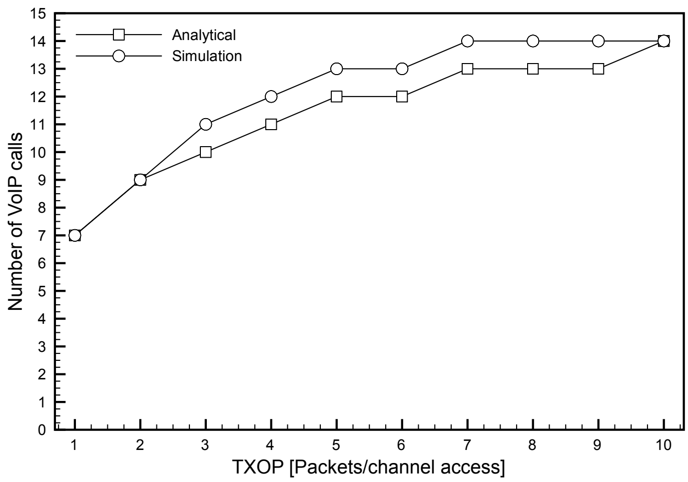

Figure 3.12: Analytical and simulation results for G.729 voice calls with a 10 ms sampling rate and increasing values of TXOP parameter

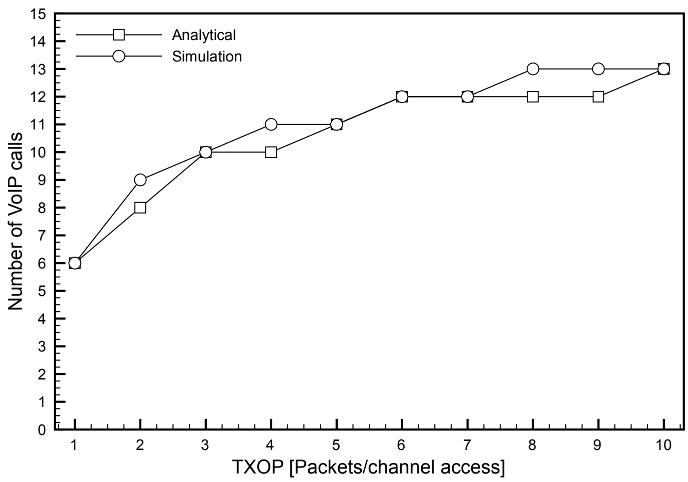

Figure 3.13: Analytical and simulation results for G.711 voice calls with a 10 ms sampling rate and increasing values of TXOP parameter

We have also considered an IEEE 802.11a/g WLAN, where the maximum bandwidth is set to 54MBit/s. Due to the increased bandwidth, the average packet service time is reduced, because a) the transmission time of the packet (T_p) is reduced, and b) MAC and system parameters such as $\sigma\epsilon$and T_{AIFS} also have different values compared with an IEEE 802.11b WLAN. In Tables 3.4

and 3.5 we show the maximum number of VoIP calls an IEEE 802.11a/g WLAN with a 54MBit/s data rate can support for selected values of TXOP parameter.

$\eta\epsilon$	Voice codec G.729, 10 ms	G.711, 10 ms	G.729, 20 ms	G.711, 20 ms
1	21	20	42	38
5	44	41	87	77
10	56	52	111	98

Table 3.4: Number of voice calls in an IEEE 802.11a WLAN with a 54 MBit/s data rate and selected values of TXOP parameter, obtained analytically

$\eta\epsilon$	Voice codec G.729, 10 ms	G.711, 10 ms	G.729, 20 ms	G.711, 20 ms
1	22	21	44	39
5	46	43	92	81
10	60	56	119	103

Table 3.5: Number of voice calls in an IEEE 802.11g WLAN with a 54 MBit/s data rate and selected values of TXOP parameter, obtained analytically

It can be observed that an IEEE 802.11g WLAN can support a slightly larger number of voice calls than an IEEE 802.11a WLAN. This is because there is a difference in the $SIFS\epsilon$ and $AIFS\epsilon$ parameters between the two protocols. In IEEE 802.11a WLAN, the SIFS is 16 μs, and the AIFS is 34 μs, whereas in an IEEE 802.11g WLAN the SIFS is only 10 $\mu s\epsilon$ and the AIFS is 28 μs. The results show that even though an IEEE 802.11a/g WLAN has approximately five times the bandwidth of an IEEE 802.11b WLAN, the number of calls do not increase by the same factor. The increase in call capacity using an IEEE 802.11a/g WLAN is between three and four times that of an IEEE 802.11b wireless network. This supports the claim that the channel access rather than bandwidth limits the voice capacity. Here the results also confirm that a significant capacity increase can be achieved when the $TXOPLimit\epsilon$ parameter is used.

Even though setting a larger TXOP parameter at the AP can improve significantly the voice capacity, the maximum number of voice calls is nevertheless limited. Figure 3.14 shows the asymptotic value for the number of voice calls when $\eta\epsilon \gg 1$ obtained analytically. For example, using G.729 voice codec with 10 ms sampling rate, this asymptotic value is 16, in an IEEE 802.11b WLAN. The actual achievable voice capacity, however, is less than this asymptotic value. It is because increasing the TXOP value will cause the bottleneck to shift from the AP to the wireless nodes. Once this happens, the wireless nodes have to wait an extended period of time before gaining channel access which then results in long delays and excessive packet loss. This is because the AP can transmit packets for the duration of $T\epsilon XOP\epsilon Limit$. For an extended $T\epsilon XOP\epsilon Limit\epsilon$duration, packets arriving in the send-queue of the wireless voice nodes will be queued, thus increasing the delay, and at some stage buffer overflows will occur. In particular, we have identified that the bottleneck shift occurs when $\eta > C_1$ where C_1 is the number of calls which can be accommodated with a default TXOP value ($\eta\epsilon = 1$).

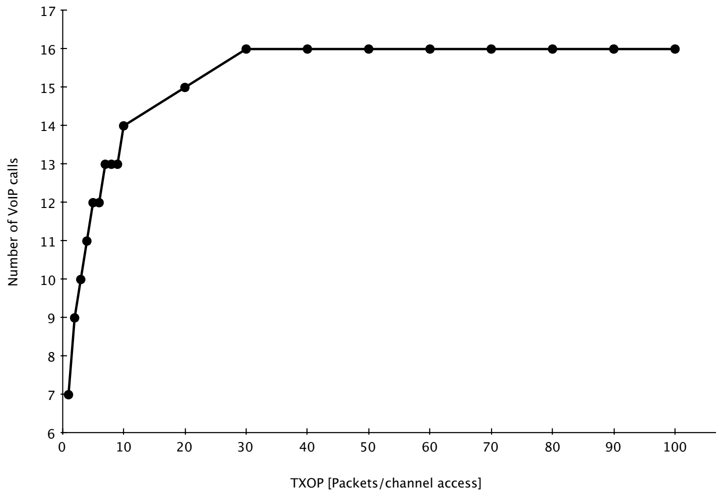

Figure 3.14: Number of G.729 voice calls (asymptotic) with a 10 ms sampling rate for large values of TXOP parameter, obtained analytically

The shift of the bottleneck at some large TXOP value is demonstrated by monitoring the average loss and delay that a wireless node experiences with increasing TXOP value. The average packet loss of the AP and a wireless node for increasing values of the TXOP parameter is shown in Fig. 3.15. It can be seen that for small values of the TXOP parameter, e.g. $\eta\epsilon = 5$, only the AP experiences packet loss (on the downlink). In contrast, when $\eta\epsilon = 10$, it is shown that the wireless node experiences excessive packet loss (on the uplink) before the AP starts to lose any of its packets. The discrepancy in packet loss between the analytical and simulation results is due to the use of the $M/M/1/K\epsilon$ model rather than a model with periodic arrivals and deterministic service time. The bottleneck shift also contributes to the difference between analysis and simulation seen in Fig. 3.15. However, note that despite this inaccuracy, the voice capacity estimate remains reasonably accurate as long as the loss is less than 2%.

As shown in Fig. 3.16 the average end-to-end network delay experienced by the wireless nodes. Observe that for $\eta < C_1$ the average end-to-end network

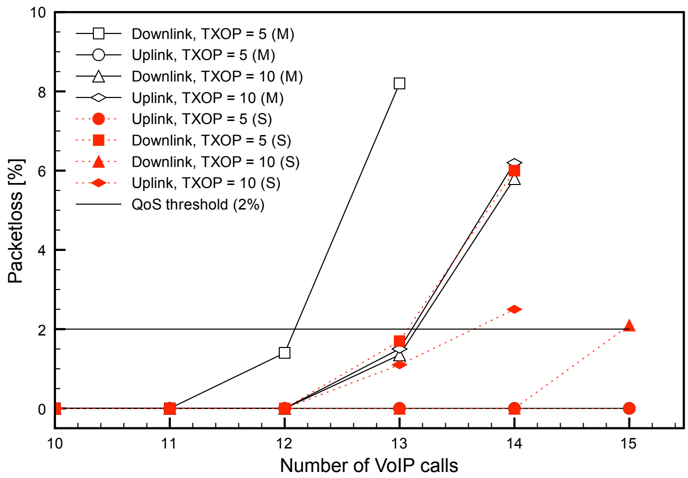

Figure 3.15: Average packet loss experienced by the AP and a wireless node for selected values of TXOP parameter and a G.729 codec with a 10 ms sampling rate (M = analytical model, S = ns-2 simulation)

delay as seen by the wireless nodes (uplink) is small. For $\eta\epsilon = 10$ however, this delay rapidly increases and exceeds a maximum delay bound, beyond which the quality of a voice call can not be maintained. Here the delay bound is set to 150 ms as in [73]. The observation that with increasing TXOP value the wireless nodes experience excessive loss and long delay indicates that the AP is no longer the bottleneck of the network. It also shows that $\eta\epsilon = C_1$ is an optimal parameter setting to obtain the maximum voice capacity in IEEE 802.11 WLAN, as $\eta\epsilon = C_1$ maximizes the voice capacity without compromising the overall performance in the WLAN, i.e. long delays.

3.6.4 VoIP capacity improvements using CW_{min}

In this section, we briefly discuss the impact of a smaller minimum contention window at the AP on the voice capacity in WLAN. Recall, that the CW_{min} parameter defines how frequently a station can attempt to transmit. This is because, a station has to wait a shorter period of time (backoff) before it can contend for the channel, thus allowing a more frequent channel access. Due to the increased channel access frequency, on average, the queue utilization is reduced and hence the contention in the WLAN is reduced. In Figs. 3.17 and 3.18 we show the number of G.729 and G.711 voice calls obtained analytically with different sampling rates when preference to the AP is given using a smaller CW_{min} parameter setting and $\eta\epsilon = 1$. As shown, the voice capacity increases by one call when the minimum contention window is half of its

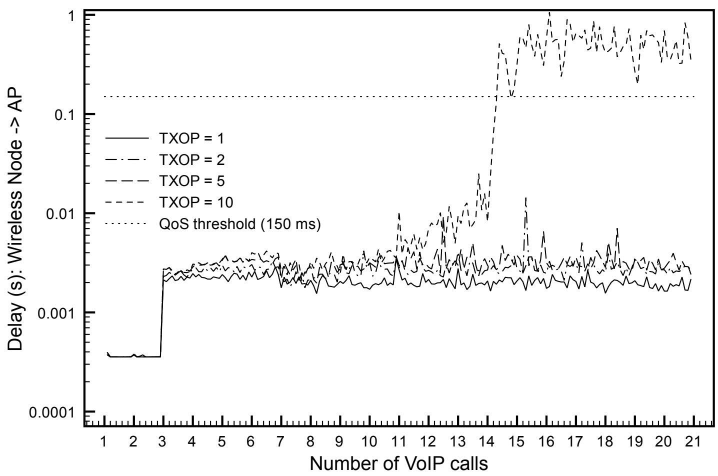

Figure 3.16: Average uplink delay experienced by the wireless nodes for selected values of TXOP parameter (ns-2 simulation) and a G.729 codec with a 10 ms sampling rate

default value. In most cases, further decreasing CW_{min} will not increase the voice capacity. This shows that even though CW_{min} can be used in conjunction with $TXOPLimit$, using our model we find that only a marginal increase in voice capacity is achieved when both parameters are used to give the AP advantages in accessing the channel. The results concerning CW_{min} are in line with previous results reported in [122, 125, 130, 145].

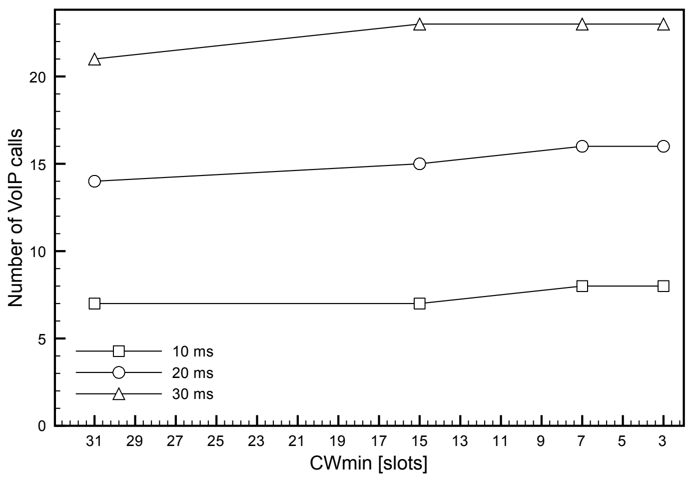

Figure 3.17: Number of G.729 voice calls obtained analytically for different sampling rates, different CW_{min} parameter and $\eta = 1$

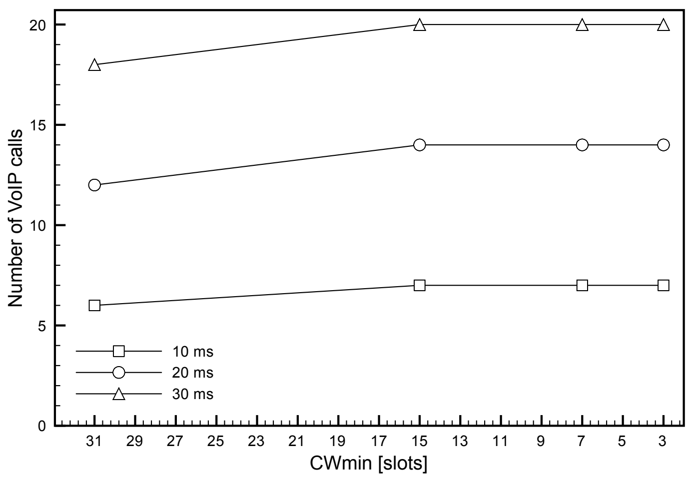

Figure 3.18: Number of G.711 voice calls obtained analytically for different sampling rates, different CW_{min} parameter and $\eta = 1$

3.6.5 VoIP capacity improvements using $AIFS$

Similar to the contention window size, the $AIFS$ parameter can be used to influence the frequency with which a station can attempt to use the channel. The $AIFS$ parameter specifies the duration the channel has to be sensed idle before attempting to transmit. By choosing a smaller than default value for the $AIFS$, the channel needs to be sensed idle for a shorter period of time, thus allowing a slightly increased frequency for the channel access.

In this scenario we use a different $AIFS$ parameter setting at the AP only to investigate if a performance gain in terms of the number of VoIP calls can be achieved. Here we use simulation to show the VoIP capacity gain, because our analytical model in Section 3.6 does not capture traffic prioritization using different $AIFS$ parameters for the AP and the wireless nodes.

In Fig. 3.19 we show the number of VoIP calls an IEEE 802.11b infrastructure can maintain for different settings of the $AIFS$ parameter. As shown, the VoIP capacity is marginally increased, e.g. one G.729 voice call for an $AIFS$ duration of $30\mu s$. The slight increase in capacity is because the AP can attempt to transmit to the channel before any wireless node can attempt to transmit. Due to the shorter period the channel has to be sensed idle, the AP can, on average, increase its transmission rate, thus reducing the queue utilization and as a result, the VoIP capacity increases. Observe that the VoIP capacity is unchanged, even if a slightly larger $AIFS$ parameter is set. This is because the $AIFS$ period has a negligible impact on the total transmission time (T_s), and the advantage the wireless voice node gain due to the increased $AIFS$ at the AP is not sufficient to reduce the call capacity.

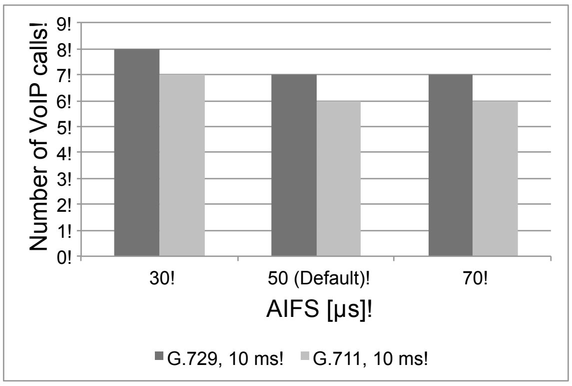

Figure 3.19: Number of G.729 and G.711 VoIP calls for different $AIFS$ parameter at the AP (ns-2 simulation)

The results show that using $AIFS$ to provide priority to the AP has only a minor impact on the overall call capacity when the WLAN carries VoIP traffic only. However, the $AIFS$ parameter can provide improved capacity if the channel is shared with other traffic types, such as TCP. By setting a smaller value of $AIFS$ for VoIP compared to TCP for example, results in a preferential channel access for the voice flows. The benefits of different $AIFS$ settings for voice traffic when the channel is shared with other traffic types has been shown in [131, 145].

3.6.6 Variable bit rate (VBR) voice streams

In our previous analysis and discussion we only considered constant bit rate voice flows. In this section we extend our work and show that the analytical model developed in Section 3.4 can also be used when the voice calls generate a variable bit rate voice stream according to the ITU P.59 [3] recommendation. We then show that a voice capacity beyond the base capacity can be achieved when an increased value of $TXOPLimit$ is set at the AP. Here we consider that the voice codecs are based on the G.729 Annex B and the G.711 Appendix II standard. It is because both extensions specify discontinued transmission (DTX), thus resulting in a variable bit rate voice stream. In Section 2.1 we discussed variable bit rate voice codecs and we outlined that during a regular conversation between two parties A and B, there are times when A is talking and B is silent (talk-spurt), when B is talking and A is silent (talk-spurt), when both parties are silent (mutual silence), and when A and B talk at the same time (double talk). This "on-off" behavior has been studied and analyzed and the ITU P.59 standard [3] provides a recommendation for an artificial conversational speech model. In Table 3.6 we show the parameters for the four different scenarios outlined above.

Parameter	Duration (s)	Rate (%)
Talk-spurt	1.004	38.53
Pause	1.587	61.47
Double talk	0.228	6.59
Mutual silence	0.508	22.48

Table 3.6: Temporal parameters in conversational speech (average for English, Italian and Japanese) [3]

The authors in [27] apply the above speech model to derive the voice capacity in an IEEE 802.11b WLAN. To derive the voice capacity for VBR voice traffic, the authors argue that the voice capacity can be obtained based on the voice capacity for CBR voice traffic and the activity ratio of the talk-spurt, and can then be calculated using [27]

$$\text{Number of VBR calls} = \left\lfloor \frac{\text{Number of CBR calls}}{\text{activity ratio}} \right\rfloor = \left\lfloor \frac{15}{0.39} \right\rfloor \approx 38.$$

Their results shown that the WLAN can support 38 VBR voice calls instead of 15 CBR calls when a G.711 voice codec with a 20 ms sampling rate is used. However, the results in [27] for the CBR and VBR voice capacity are only a coarse approximation of the actual voice capacity. This is because the analytical model used in [27] does not consider collisions and the obtained voice capacity is based on the packetization interval and the successful transmission time of a packet only. The collision time cannot be neglected, because with an increasing number of voice calls in the WLAN, the average collision time for the wireless nodes and the AP increases, as shown in Fig. 3.20. Because the average collision time changes the total average transmission time of the packet, the average packet rate at the AP is also changed, thus leading to a skewed voice capacity in WLAN.

The assumption in [27] that the number of VBR voice calls can be derived using the number of CBR calls induces further imprecisions. This is because, on average, a variable bit rate voice stream generates fewer packets per time unit. As a result the average packet arrival rate (λ) and hence the queue utilization (ρ) at the AP changes. Because of these changes, other internal system dynamic parameters such as the conditional collision probability and the average collision times will change, hence affecting the calculated packet loss and

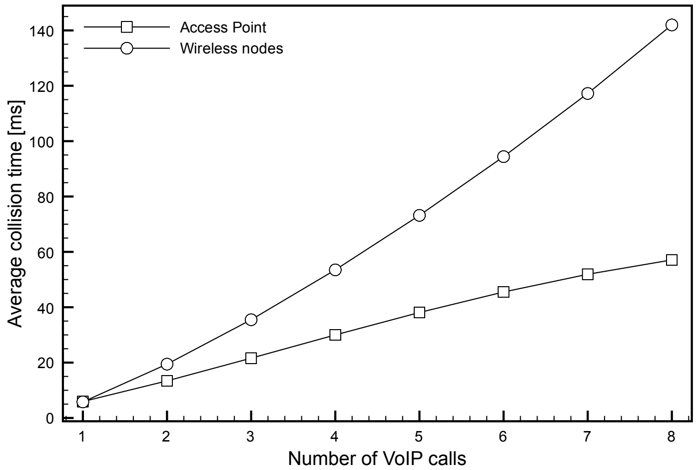

Figure 3.20: Average collision times of the access point and the wireless node for a G.729 voice codec with a 10 ms sampling rate obtained analytically

delay at the AP, leading to a difference in voice capacity.

To accurately obtain the number of VBR voice calls an IEEE 802.11 WLAN can support, let ψ denote the voice activity rate as defined in the ITU P.59 standard and given in Table 3.6. Using the simplified assumption that the voice codec only generates packets when a user is talking, in this scenario $\psi \approx 0.39 = 39\%$, then the packet arrival rate of the VBR voice traffic at a wireless node and the AP is given by

$$\widetilde{\lambda_n} = \psi\lambda_n, \tag{3.22}$$

$$\widetilde{\lambda_a} = (N-1)\psi\lambda_n. \tag{3.23}$$

Now replacing λ_n with $\widetilde{\lambda_n}$ and λ_a with $\widetilde{\lambda_a}$ in the analytical model, the voice capacity can then be obtained using Eq. (3.21) such that.

$$\begin{aligned} p_a &\approx \frac{(1-\rho_a)\rho_a^K}{1-\rho_a^{K+1}} \\ &= \frac{\left(1-\frac{\widetilde{\lambda_a}}{\mu_a}\right)\left(\frac{\widetilde{\lambda_a}}{\mu_a}\right)^K}{1-\left(\frac{\widetilde{\lambda_a}}{\mu_a}\right)^{K+1}}. \end{aligned} \tag{3.24}$$

In Fig. 3.21 we show the packet loss probability at the AP for G.729 and G.711 CBR and VBR voice calls with a 10 ms and 20 ms sampling rate and the voice activity rate ψ. As shown, whenever the voice codecs generate a variable bit rate stream, the number of voice conversations that can be maintained is significantly increased. For example, the WLAN can maintain up to 7 voice calls using the G.729 voice codec with a 10 ms sampling rate generating a CBR voice stream, but the WLAN can support up to 20 voice calls if the same voice codec generates a variable bit rate voice stream. A similar increase in voice capacity can also be observed for a 20 ms sampling rate as well as for the G.711 voice codec. We validated our results by simulation, and it can be observed that the simulation results are in agreement with those of the analytical model. Note that the simulation data is shown with a 95% confidence interval.

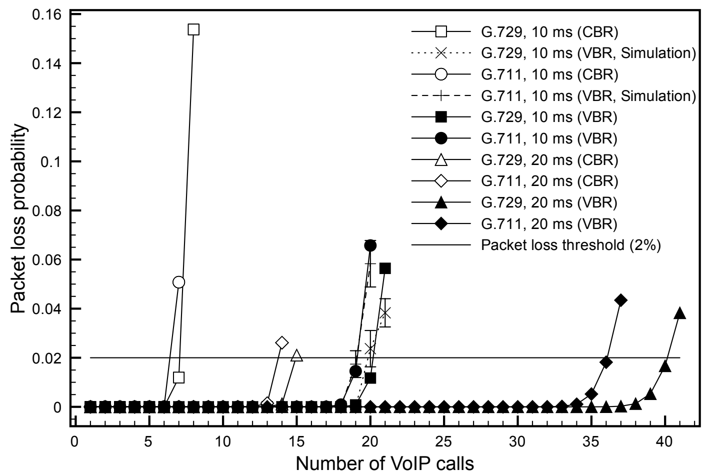

Figure 3.21: Observed packet loss for CBR/VBR G.729 and G.711 VoIP calls obtained analytically and by simulation (where indicated). Note that the VBR VoIP call sampling rate is an equivalent rate as defined in Eqs. (3.22) and (3.23)

The voice capacity increase can also be seen when a network delay is used to determine the number of VoIP calls. In Fig. 3.22 we show the network delay on the downlink when the WLAN serves G.711 and G.729 voice calls and the calls either generate a CBR or a VBR flow. Recall that every 10 s a new call is added to the WLAN.

Even though the voice capacity is increased when the voice codecs generate a variable bit rate, the AP is still the bottleneck in the WLAN, limiting the number of voice calls. To mitigate the bottleneck problem, here we also set an increased value of $T\epsilon XOP\epsilon Limit\epsilon$at the AP using our analytical model in Section 3.4 and the packet arrival rates for VBR flows as above. In Fig. 3.23 we show the number of G.729 and G.711 VBR voice calls with a 10 ms sampling rate for an increasing value of TXOP parameter. As shown, the number of G.729 voice calls increases from 20 to 25 voice calls for $\eta\epsilon = 2$, and 32 calls can be supported for $\eta\epsilon = 5$.

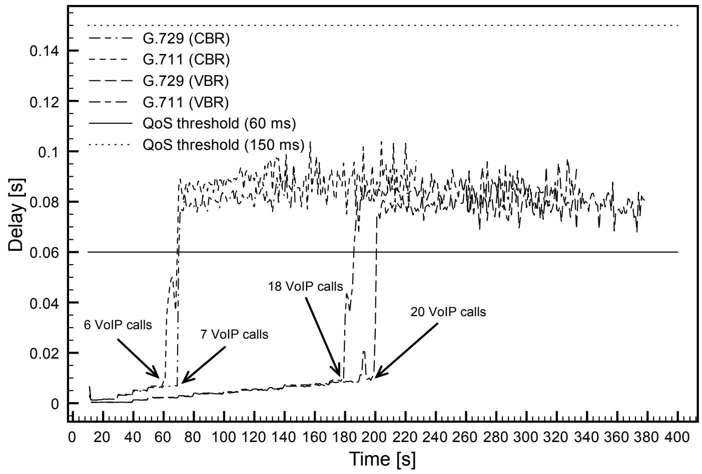

Figure 3.22: Network delay in the downlink direction for CBR/VBR voice calls using the G.711 and the G.729 voice codec (ns-2 simulation)

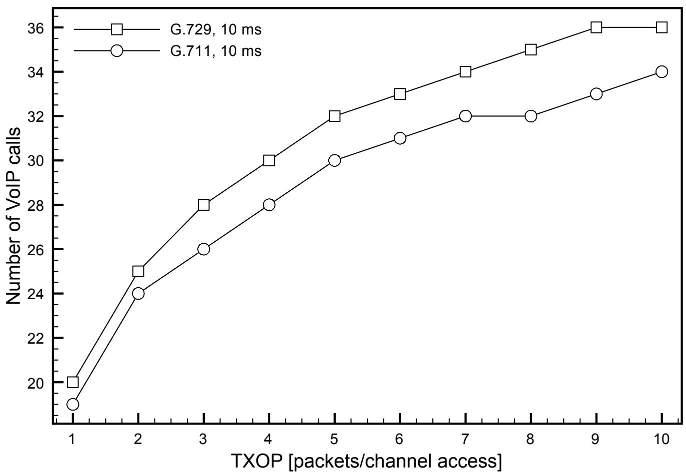

Figure 3.23: Number of G.729 and G.711 VBR voice calls obtained analytically for increasing values of TXOP parameter

Note that the difference in VoIP capacity of 2 to 3 voice calls between our results and the results presented by Shin and Schulzrinne [27] is because i) Shin and Schulzrinne [27] do not consider the average collision time in their analytical approach to obtain the number of CBR and VBR VoIP calls, and ii) the number of VoIP calls a WLAN can support is derived using network delay in their experimental approach, rather than packet loss modeling as is done in our approach. Even though the call capacity is comparable, from the modeling point of view it is important to capture the collision times as it impacts on the total transmission time and therefore may lead to some differences in results. Furthermore, recall that in Section 3.4 we argued that the AP is a bottleneck and that packet loss will be the main factor for voice call quality degradation, and that the acceptable delay depends on the users perception of what is a tolerable delay. Overall, it seems likely that these differences in approach lead to the slight difference in VoIP call capacity observed by us and Shin and Schulzrinne.

3.7 Summary

In this chapter we studied voice over IP traffic in IEEE 802.11 infrastructure wireless networks. In particular, we focused on the voice capacity that is defined as the maximum number of voice calls a WLAN can support with an adequate level of call quality. We have developed a detailed analytical model to obtain the voice capacity and confirmed that the voice capacity is severely limited. Note that our model also accurately captures the slight difference in backoff behavior between EDCA and DCF. We then evaluate a solution to improve the voice capacity based on the adjustable $T\epsilon XOP\epsilon Limit\epsilon$MAC parameter. The emphasis here was to address the AP bottleneck problem by using the adjustable $T\epsilon XOP\epsilon Limit\epsilon$parameter of the IEEE 802.11e protocol. Using our analytical model we showed the effectiveness of this solution and showed that a 100% capacity gain can be achieved. We showed that setting a larger $T\epsilon XOP\epsilon Limit\epsilon$ at the AP improves the voice capacity significantly, and we outlined that setting $\eta\epsilon = C_1$ is an optimal TXOP value. We validated our analytical results using simulation as well as WLAN testbed[8] measurements. This showed that the assumption made in [2] that $\eta\epsilon$should be equal to the number of voice calls will not hold. Additionally we showed that increasing TXOP beyond that optimal value will not increase the voice capacity. In fact, setting a larger $T\epsilon XOP\epsilon Limit\epsilon$

[8]WLAN testbed measurement are limited to $\eta = 1$ and $\eta = 2$, due to limited testbed equipment.

will affect the overall performance of VoIP in WLAN, because the bottleneck will shift to the wireless nodes. By also considering an IEEE 802.11a/g WLAN we confirmed that the channel access mechanism is limiting the voice capacity rather than the available bandwidth. To evaluate if a further capacity gain can be achieved, we also set different values of CW_{min} and/or $AIFS$ parameter at the AP. Our analysis showed however, that the additional capacity gain is only marginal. Finally, we showed that our analytical model is versatile and can also be used to obtain the number of voice calls a WLAN can support if the voice codec generates a variable bit rate stream.

4

The impact of the buffer size on the VoIP capacity

In the previous chapter we studied voice over IP traffic in IEEE 802.11 infrastructure WLANs, and we confirmed that the number of voice calls that can be supported with an acceptable level of quality is severely limited. As in an infrastructure WLAN all traffic from and to the wireless nodes have to pass through the access point (AP), acting as a common bridge between the wired and the wireless network. With an increasing number of calls the AP becomes a bottleneck, restricting the number of voice calls. We have shown that with an increasing number of voice calls in the WLAN, the number of packets that require queueing at a station increases. This is because of asymmetric channel access, i.e. the probability with which the AP and all wireless nodes gain channel access is equal. Also the traffic arrival rate at the AP is a superposition of each individual rate at the wireless node, the number of packets that require queueing at the AP increases rapidly, and at some stage exceeds the available buffer space. As a result, packet loss occurs and the voice call quality deteriorates until it is deemed to be no longer acceptable.

Following our VoIP capacity analysis, we then showed that a significant voice capacity gain can be achieved if a channel access preference is given to the AP by increasing the adjustable $TXOPLimit$ parameter at the AP only. Due to the increased $TXOPLimit$ the AP can now transmit multiple packets per channel access, thus reducing the number of packets that are queued at the AP. We showed that then a capacity gain of approximately 100% can be achieved. Nevertheless, we also showed that even setting a larger than default value of $TXOPLimit$ at the AP will at some stage cause packet loss, as the buffer is exceeded. Furthermore, we also showed that setting a too large value of the $TXOPLimit$ parameter will impact on the overall performance of the WLAN and can cause a gradual bottleneck shift from the AP to the wireless nodes.

Even though we showed that an access preference at the AP can improve the VoIP capacity in the WLAN, if the number of calls keeps increasing, at some stage the AP experiences packet loss above the defined QoS threshold because the packet arrival rate at the AP exceeds the combined service rate and buffering at the AP. In this chapter we investigate the impact of different buffer sizes at a station on the VoIP capacity in IEEE 802.11 infrastructure wireless networks, because one might assume that the VoIP capacity is increased if a station has a larger buffer size. However, we show that this is not the case. In particular, we show that there exists a buffer size with which the voice capacity is maximum, and further increasing the buffer size will not result in an increased voice capacity. We then show that this finding also holds for different QoS parameter settings at the AP used to mitigate the bottleneck problem.

The remainder of this chapter is organized as follows: In Section 4.2 we analyze the impact of the buffer size on the number of voice calls that can be supported with an acceptable level of quality. We then study the buffer requirements when the adjustable MAC parameters are changed to given preferential channel access to the AP. In Section 4.4 we discuss the impact of variable buffer space at the wireless node, and in Section 4.5 we provide an insight into the buffering requirements for variable bit rate voice streams. Based on our findings in Section 4.2, in Section 4.6 we develop a new analytical model and derive a closed-form expression for the voice capacity in IEEE 802.11 infrastructure WLANs. This is followed by the validation of the closed form expression in Section 4.6.1. Based on our closed-form expression for the voice capacity, in Section 4.7 we derive a novel way of obtaining the voice capacity in WLAN and propose a VoIP capacity approximation formula that can be used in conjunction with the adjustable $TXOPLimit$ parameter. Here we will also discuss some further insights we can gain from this approximation. We conclude the chapter in Section 4.8 with a summary of findings and contributions.

Our contributions in this chapter can be summarized as follows:

1. We show that there exists a minimum buffer size (K_{min}) with which the voice capacity in a WLAN is maximum and further increasing the buffer will not increase the number of VoIP calls a WLAN can support.

2. We show that the voice capacity is independent of the buffer size for $K \geq K_{min}$ and we show that therefore an $M/G/1/\infty$ queueing model can be used to obtain the voice capacity.

3. Based on the $M/G/1/\infty$ queueing model we provide a closed form expression for the number of voice calls a WLAN can support .

4. We show that the minimum buffer size K_{min} with which the voice capacity is maximum also holds for variable bit rate voice streams.

5. We show that the effect of the bottleneck shift can be reduced if different buffer values are used at the wireless nodes.

6. We propose a novel way of obtaining the voice capacity in infrastructure WLAN, when preference is given to the AP using the $TXOPLimit$ parameter, and we show some additional insight that can be gained for this novel VoIP capacity approximation formula.

4.1 Overview & Approach

The impact of the buffer size in IEEE 802.11 WLANs has previously been studied in [2, 25, 158, 183]. However, whereas [158] is concerned with TCP traffic in IEEE 802.11e WLANs, and the authors in [183] investigate whether different buffer sizes can be used to guarantee a fair channel access, Dangerfield et al. [2] and Malone et al. [25] investigate the impact of different buffer sizes on the VoIP performance.

In [25], Malone et al. investigate the impact of small buffer sizes on the voice capacity in an infrastructure WLAN. In particular, Malone et al. consider very small buffer sizes of 1, 2, 5 and 10 packets at the AP and a wireless station. They showed that with an increased buffer size at the AP, the throughput can be increased, and that an increased number of calls can be supported with a larger buffer. Similar results are presented in [2], where Dangerfield et al. use testbed measurements to study the voice performance for variable buffer space.

Furthermore, Dangerfield et al. use the adjustable CW_{min} and $TXOPLimit$ MAC parameter to increase the throughput, decrease delay and subsequently improve the number of concurrent voice conversations. Whereas in [25] Malone et al. concluded that the rapid transition from a low-loss, low-delay environment to a high-loss, high-delay environment is insensitive to the buffer size, in [2] it is suggested that the buffer size at the AP should be proportional to the number of VoIP calls. However, our findings indicate a different insight when a variable buffer size is used at a station.

To analyze the impact different buffer size have on the voice capacity in a WLAN, we use our analytical model defined in the previous chapter. Recall that our analytical model is based on an $M/G/1/K$ queueing model, where K is the queue size in packets. Even though there are many different types of queues [184] and queueing procedures, e.g. FIFO (first-in first-out) or LIFO (Last-in first-out) , here we only consider a packet based queue and FIFO.

4.2 Buffer size analysis

Similar to Chapter 3, we first investigate the number of voice calls that can be supported when the default TXOP value ($\eta = 1$) is used with different values of the buffer size K. Recall that in our analysis in Chapter 3 we set the buffer at a station to $K = 50$ packets.

In Fig. 4.1 we show the number of G.729 voice calls with a 10 ms sampling rate that can be supported for buffer sizes of 10, 30, 50 and 100 packets, obtained using our analytical model and simulation. It can be seen that for buffer sizes of $K = 50$ and $K = 100$ packets the number of voice calls does not change, and the WLAN can support 7 voice calls. This shows that even doubling the buffer size at a station does not affect the number of calls that can be maintained.

Reducing the buffer size to $K = 30$, it is shown that there is a one call difference between the analytical and simulation results, however, both methods still suggest that the number of G.729 voice calls is 7, or 6 calls as reported by simulation. A change in voice capacity can be observed if the buffer size is set to $K = 10$. Due to the limited amount of buffer space, the queue reaches saturation quicker, and as a result packet loss occurs, and then the voice capacity is reduced to 6 calls.

In Table 4.1 we show the maximum number of G.729 and G.711 voice calls with a 10 ms and 20 ms sampling rate for different buffer sizes K, obtained

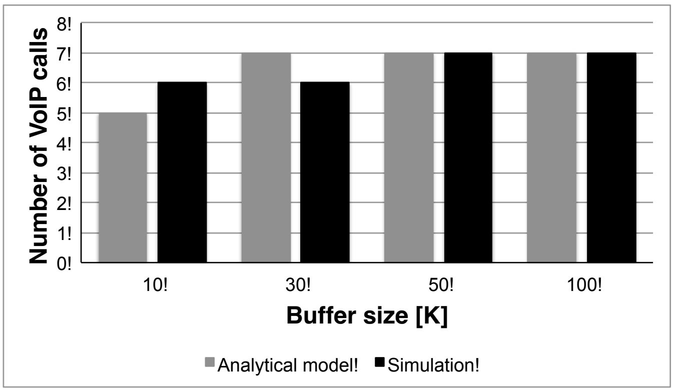

Figure 4.1: Number of G.729 voice calls with a 10 ms sampling rate with different values of the buffer size K, obtained analytically

analytically. Observe that there exists a minimum buffer size ($K_{min} = 30$ packets) with which the VoIP capacity is a maximum, and increasing $K\epsilon$beyond K_{min} will not result in an increased number of voice calls. It can also be seen that for buffer size $K < K_{min}$ the VoIP capacity is decreased. This is because the small buffer size is insufficient to cope with the difference in packet arrival and packet service rates at the AP.

$K\epsilon$	G.729, 10 ms	G.711, 10 ms	G.729, 20 ms	G.711, 20 ms
10	5	5	11	10
20	6	6	13	12
30	7	6	14	12
40	7	6	14	12
50	7	6	14	12
100	7	6	14	13
500	7	6	14	13
1000	7	6	14	13

Table 4.1: Number of G.729 and G.711 voice calls obtained analytically for different buffer sizes K in an IEEE 802.11b infrastructure WLAN

In Fig. 4.2 it is shown analytically that increasing the buffer beyond K_{min} reduces the packet loss at the AP slightly, however, the overall VoIP capacity does not change. This also confirms our finding that K_{min} is the minimum buffer size that maximizes the VoIP capacity.

The optimality of the above K_{min} buffer size can also be observed by examining the network delay (obtained by simulation), as shown in Fig. 4.3. As shown, for $K_{min} = 30$ packets, the network delay in the downlink direction experienced by the wireless VoIP nodes is below the strict delay threshold of 60 ms [27] and well below the 150 ms delay bound applied in [73], while maximizing the number of VoIP calls. Observe that the experienced average network

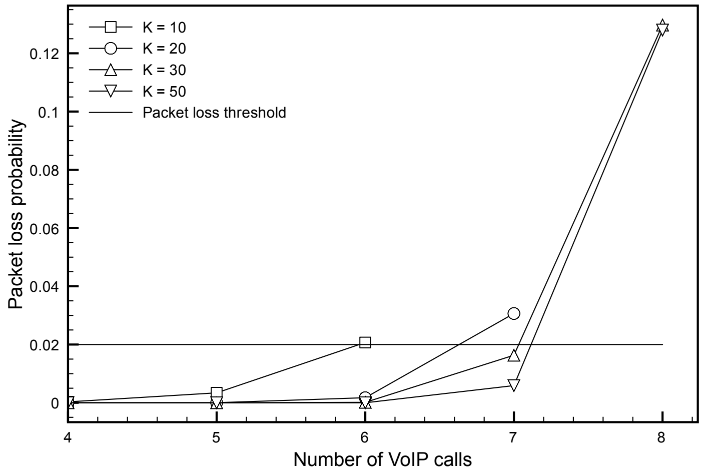

Figure 4.2: Packet loss probability for G.729 voice calls with a 10 ms sampling rate at the AP, with $\eta = 1$ and different buffer sizes K obtained analytically

delay increases with increasing buffer sizes. This is not unexpected, because the average time a packet is queued is increased as more packets can be queued at the AP. Nevertheless, for $K\epsilon = K_{min} = 30$ packets the delay is below very strict delay bounds, and as shown above, also maximizes the number of VoIP calls if packet loss is used to derive the VoIP capacity. Note however, that using delay to derive the VoIP capacity depends on the user's perception of what is a tolerable delay.

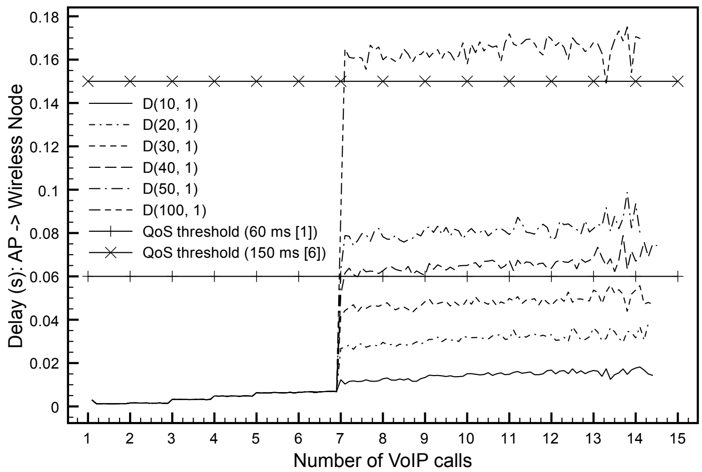

Figure 4.3: Downlink network delay for G.729 voice calls with a 10 ms sampling rate with $\eta = 1$ and different buffer sizes K [D(K, η)] (ns-2 simulation)

In Tables 4.2 and 4.3 we show the number of calls in an IEEE 802.11a and IEEE 802.11g WLAN for different buffer size $K\epsilon$obtained analytically. It can be seen that our observation of K_{min} also holds even if a different, higher bandwidth PHY is used.

$K\epsilon$	G.729, 10 ms	G.711, 10 ms	G.729, 20 ms	G.711, 20 ms
10	18	17	35	32
20	20	19	40	36
30	20	19	41	37
40	21	20	42	38
50	21	20	42	38
100	21	20	42	38
500	21	20	42	38
1000	21	20	42	38

Table 4.2: Number of G.729 and G.711 voice calls obtained analytically for different buffer sizes K in an IEEE 802.11a infrastructure WLAN

$K\epsilon$	G.729, 10 ms	G.711, 10 ms	G.729, 20 ms	G.711, 20 ms
10	18	17	37	33
20	21	20	41	37
30	21	20	43	39
40	22	20	44	39
50	22	21	44	39
100	22	21	44	39
500	22	21	44	39
1000	22	21	44	39

Table 4.3: Number of G.729 and G.711 voice calls obtained analytically for different buffer sizes K in an IEEE 802.11g infrastructure WLAN

4.3 Buffer requirements for adjustable IEEE 802.11 MAC parameter

4.3.1 Effects of different $TXOPLimit$ parameter values

In Chapter 3 we showed that a significant voice capacity gain can be achieved when the adjustable $T\epsilon XOP\epsilon Limit\epsilon$ parameter of the IEEE 802.11 medium access control is used to give a channel access preference to the AP. In this section we will investigate the buffer requirements for a station when a channel access preference is given to the AP as in Chapter 3. In particular we are interested in the finding of the minimum buffer (K_{min}) discussed in the previous section also applies in this scenario, because one might argue that a smaller than $K_{min} = 30$

packets buffer size is required at the AP, due to the transmission of multiple packets per channel access.

In Tables 4.4 and 4.5 we show the number of voice calls that a WLAN can maintain before the packet loss threshold of 2% is exceeded for different buffer size $K\epsilon$and selected values of the $TXOPLimit\epsilon$parameter $\eta\epsilon$using our analytical model developed in Chapter 3 and simulation. It can be observed that even if the variable $TXOPLimit\epsilon$parameter is used, the minimum buffer size K_{min} is still sufficient to maximize the number of voice calls that can be maintained, and that a further increase in buffer size has no impact on the overall voice capacity. Also note that even though the AP can transmit multiple packets per channel access, setting $K < K_{min}$ reduces the VoIP capacity by 1 to 2 calls, irrespective of the TXOP value η. This finding further supports our claim that $K_{min} = 30$ packets is an optimal buffer size for VoIP in WLANs.

$\eta\epsilon$	1		2		5		7	
$K\epsilon$	M	S	M	S	M	S	M	S
10	5	6	7	9	10	11	10	11
20	6	6	8	9	11	12	12	13
30	7	6	9	9	12	12	13	13
40	7	7	9	9	12	13	13	13
50	7	7	9	9	12	13	13	14
100	7	7	9	9	12	13	13	13

Table 4.4: Number of G.729 voice calls with a 10 ms rate for different buffer sizes K and selected values of the TXOP parameter η (M = analytical model, S = ns-2 simulation)

$\eta\epsilon$	1		2		5		7	
$K\epsilon$	M	S	M	S	M	S	M	S
10	5	6	7	8	9	10	10	11
20	6	6	8	8	10	11	11	12
30	6	6	8	9	11	12	12	12
40	6	6	8	9	11	11	12	12
50	6	6	8	8	11	11	12	12
100	6	6	9	9	11	11	12	12

Table 4.5: Number of G.711 voice calls with a 10 ms rate for different buffer sizes K and selected values of the TXOP parameter η (M = analytical model, S = ns-2 simulation)

4.3.2 Effects of different CW_{min} parameter

Whereas in the previous section we only modified the adjustable $T\epsilon XOP\epsilon Limit\epsilon$ parameter, here we confirm that our finding of the minimum buffer size K_{min} holds when the $T\epsilon XOP\epsilon Limit\epsilon$ parameter and the contention window sizes are both adjusted.

As we have previously shown in Section 3.6.4 adjustments to the CW_{min} and CW_{max} parameter only marginally affect the voice capacity in WLANs. In Figs. 4.4 and 4.5 we show the number of voice calls that a WLAN can support for different buffer sizes $K\epsilon$(10, 30, 50 and 100 packets), selected values of $T\epsilon XOP\epsilon Limit\epsilon$ parameter $\eta\epsilon$(1, 2, 5 and 7 packets per channel access) as well as different settings for the CW_{min} parameter (7, 31 and 127 slots). It can be seen that reducing the contention window size to a smaller than default value ($CW_{min} = 31$) will marginally increase the voice capacity. Irrespective of the increase however, the buffer limit still applies, and it can be seen that the maximum voice capacity is reached for $K\epsilon = 30$ packets, for all configurations of CW_{min} and η. Similar to our previous results, increasing the contention window to a value larger than the default will reduce the voice capacity. This is because the average backoff time is increased, thus increasing the average service time of a packet, and as a result increases queueing delay of packets at the AP. Similar to the previous case where the contention window was smaller than default, here the K_{min} finding still applies, and the maximum number of calls is reached for $K\epsilon = 30$ packets. The results here show that our finding still holds even if $T\epsilon XOP\epsilon Limit\epsilon$ and CW_{min} parameters are varied.

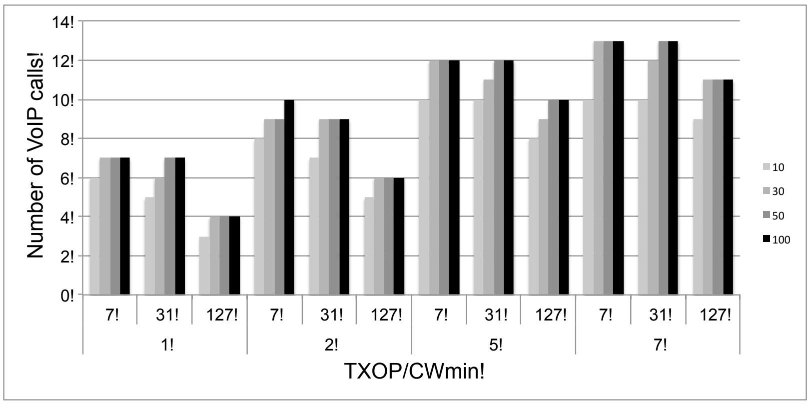

Figure 4.4: Number of G.729 voice calls with a 10 ms sampling rate obtained analytically for selected values of η, different buffer sizes K and CW_{min} parameter

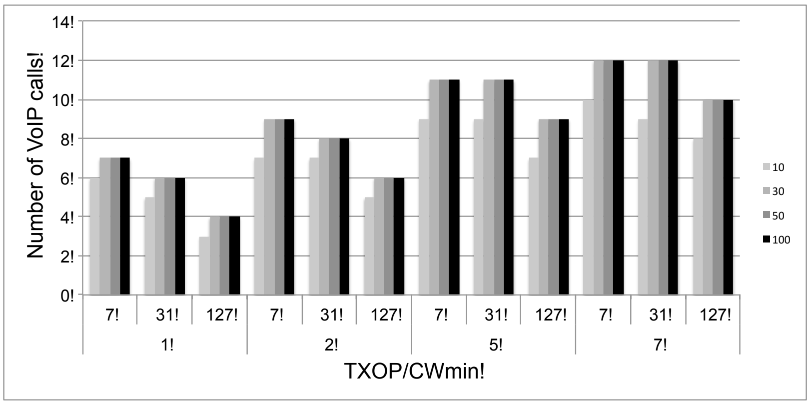

Figure 4.5: Number of G.711 voice calls with a 10 ms sampling rate obtained analytically for selected values of η, different buffer size K and CW_{min} parameter

4.3.3 Effects of different $AIFS$ parameter

In Section 3.6.5 we have shown that changes to the arbitrary inter-frame space ($AIFS$) parameter have only a small impact on the VoIP capacity, e.g. a capacity increase by one call for an $AIFS$ value of $30\mu s$. In Table 4.6 we show the number of G.729 and G.711 VoIP calls obtained by simulation for different values of $AIFS$ parameter and different buffer sizes K. As shown, VoIP capacity is increased by one call for $AIFS = 30\mu s$. Also observe that our minimum buffer size K_{min} holds, e.g. for a G.729 voice codec with a 10 ms sampling rate and a smaller value of the $AIFS$ parameter. The results nevertheless confirm our previous finding that the overall impact of the $AIFS$ parameter, to give priority to the AP only, is negligible.

	G.729, 10 ms			G.711, 10 ms		
$AIFS$ / K	30 μs	50* μs	70 μs	30 μs	50* μs	70 μs
10	7	6	6	7	6	6
30	8	6	6	7	6	6
50	8	7	7	7	6	6
100	8	7	7	7	6	6

Table 4.6: Number of G.729 and G.711 VoIP calls with a 10 ms sampling rate for different buffer sizes K and selected values of $AIFS$ parameter (* default parameter) (ns-2 simulation)

4.4 Variable buffer space at a wireless node

In our analysis of the buffer size and its potential impact on the voice capacity in WLAN we considered that changes to the buffer size K were made at a station, that is the AP and/or the wireless nodes. We showed that there is a minimum buffer size that maximizes the voice capacity and further increasing the buffer will not increase the voice capacity. The assumption of setting all buffer sizes equal at all stations is acceptable, because we have shown that the limited channel access at the AP limits the VoIP capacity, rather than the limited buffer size. Also, in Section 3.6.2 we have shown that a wireless VoIP node, on average, only queues a small number of packets. In this section, we investigate the impact of different buffer sizes at the wireless VoIP nodes, and apply the default buffer size ($K\epsilon = 50$ packets) at the AP.

Setting different buffer size $K\epsilon$at the wireless nodes can affect the performance of VoIP in wireless networks. In Chapter 3 we have shown that setting too large values of $T\epsilon XOP\epsilon Limit\epsilon$parameter $\eta\epsilon$at the AP will cause a bottleneck shift to the wireless nodes. This is because with large values of η, the wireless nodes have to wait an extended period of time before they can attempt to transmit a packet. Due to the extended waiting period, an increased number of packets are queued at a node, and at some stage may exceed the queue size. As a result, the wireless nodes experience packet loss and long delays in the uplink direction.

By setting a larger buffer at the wireless nodes ($K_n > K_a$) the bottleneck shift can be delayed. This is because the wireless nodes can now queue an increased number of packets, and thus packet loss only occurs if $\eta\epsilon \gg \not C_1$. Setting a larger buffer at the wireless nodes has no impact in a low traffic WLAN, i.e. two to three voice calls, however, whenever the WLAN is close to saturation the experienced delay is increased significantly, and depending on the tolerable delay, the voice quality may not be acceptable, as shown in Fig. 4.6. Recall, that during the simulation a VoIP call is added to the WLAN every 10s, e.g. at time $T\epsilon = 30s$, the WLAN carries 3 full-duplex VoIP calls, at time $T\epsilon = 100s\epsilon$ the WLAN carries 10 full-duplex VoIP calls, and so forth.

For example, we have shown that for $\eta\epsilon = 10$ the WLAN can maintain 14 voice calls using a G.729 voice codec with a 10 ms sampling rate, before the packet loss in the downlink direction exceeds the QoS threshold. However we have also shown, that for this configuration, the wireless nodes experience packet loss before the AP loses any packets, thus indicating a bottleneck shift has occurred. By setting a smaller buffer size at the wireless node (K_n), the wireless nodes can store fewer packets for the same configuration, and packet loss occurs earlier, as shown in Fig. 4.7 for $K_n = 10$. On the other hand setting a larger buffer at the wireless nodes compared to the AP, may delay

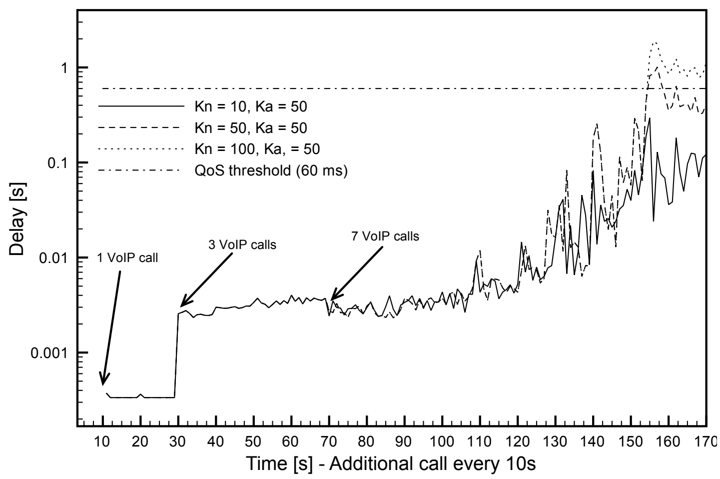

Figure 4.6: Uplink delay of G.729 voice calls with a 10 ms sampling rate, different buffer size K at the wireless nodes and $\eta = 10$ (ns-2 simulation)

the bottleneck shift. As shown in Fig. 4.7, if the wireless nodes have a buffer $K\epsilon = 200$, the wireless nodes experience packet loss when the 16th call joins the WLAN. However, the AP already exceeds the QoS threshold when the 15th call joins. This shows that setting larger buffer sizes at a wireless node can delay the bottleneck shift.

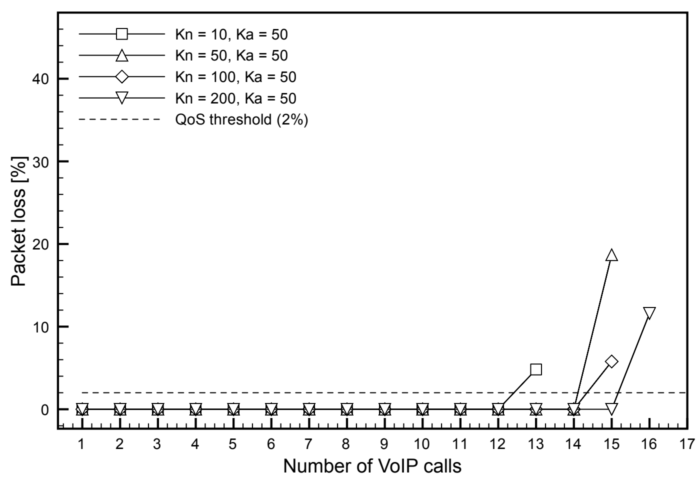

Figure 4.7: Uplink packet loss of G.729 voice calls with a 10 ms sampling rate, different buffer size K at the wireless nodes and $\eta = 10$ (ns-2 simulation)

These simulation results strongly suggest that the K_{min} parameter is not only the minimum buffer size required to maximize the voice capacity, but it is also the ideal value without compromising the overall performance. Overall this provides strong evidence that our findings of K_{min} for the buffer size and $\eta\epsilon = C_1$ for the $T\epsilon XOP\epsilon Limit\epsilon$ parameter are optimal for an IEEE 802.11 infrastructure WLAN.

4.5 Buffer requirements for VBR voice streams

Similar to our analysis in Chapter 3, here we also extend our evaluation to variable bit rate voice streams and investigate the buffer requirements for these streams. In particular, we focus on the minimum buffer size required to maximize voice capacity.

As we have shown in the previous sections of this chapter, for constant bit rate voice streams there is a minimum buffer size $K_{min} = 30$ packets that is required to maintain the maximum number of voice calls an IEEE 802.11b infrastructure WLAN can support. Using our analytical model developed in Chapter 3 and the approach for VBR streams as discussed in Section 3.6.6, here we apply different settings of the buffer size $K\epsilon$ to investigate the buffer size requirements for VBR voice streams.

In Table 4.7 we show the maximum number of voice calls using a G.729 and G.711 voice codec each with a 10 ms sampling rate for selected value of the $T\epsilon XOP\epsilon Limit\epsilon$ parameter $\eta\epsilon$ and variable buffer size K. As shown in the previous chapter, for the default TXOP value $\eta\epsilon = 1$ and the default buffer size of $K\epsilon = 50$ packets, the wireless network can support up to 20 and 18 voice calls using the G.729 and G.711 voice codec, respectively. The results show that a small buffer size, e.g. $K\epsilon = 10$ will reduce the number of calls that can be supported. It can also be observed that setting a larger buffer than default, here 100 packets, does not yield an increase in call handling capacity. These results are in line with those presented in Section 3.6.6 for constant bit rate voice traffic. However, the results shown in Table 4.7 also show that the same minimum buffer size K_{min} applies for a VBR stream as it does for a CBR voice stream with a wide range of different $T\epsilon XOP\epsilon Limit\epsilon$ parameter settings. This additionally supports our finding that K_{min} is the optimal buffer size in a wireless LAN when the network carries voice flows.

4.6 An analytical model with infinite buffer space

As we have shown in the previous sections of this chapter, that there exists a minimum buffer size $K_{min} = 30$ packets that is required to support the maximum number of voice calls which can be maintained with an acceptable level

$\eta\epsilon$	1		2		5		7	
$K\epsilon$	G.729	G.711	G.729	G.711	G.729	G.711	G.729	G.711
10	17	16	21	20	26	24	28	26
20	19	18	24	22	30	28	32	30
30	20	18	24	23	32	30	33	31
40	20	18	24	23	32	30	34	32
50	20	19	25	23	32	30	34	32
100	20	19	25	23	32	30	34	32

Table 4.7: Comparison of the maximum number of G.729 and G.711 voice calls with a 10 ms sampling rate for different buffer sizes K and selected values of TXOP parameter η obtained analytically

of quality, with packet loss below 2% and delay less than the strict delay bound of 60 ms [27]. We also showed that increasing the buffer size beyond K_{min} will not provide further benefits in increasing the voice capacity. In particular we showed that even if an arbitrarily large buffer is set e.g. 500 or 1000 packets, the number of calls supported does not increase.

Based on the findings, we claim that the maximum number of voice calls in an IEEE 802.11 infrastructure WLAN is independent of the buffer size, given the buffer is at least K_{min}. To confirm our claim, we modify our model developed in Chapter 3 such that $K\epsilon = \infty$, and therefore the VoIP capacity can be derived using an $M/G/1/\infty$ queueing system to model the AP. In this model, there is no packet loss and the maximum capacity is calculated based on the stability condition of the AP queue. In particular, the number of voice calls is calculated based on the inequality $\lambda_a < \mu_a$, which guarantees the queue stability. This approach is somewhat similar to [103], however, whereas Cai et al. [103] use the fixed-point to derive the required system parameter, we provide a closed form expression for the voice capacity in an IEEE 802.11 infrastructure WLAN. The closed form expression for the VoIP capacity is versatile such that it can accommodate variable parameter changes of the IEEE 802.11 medium access control, e.g. $T\epsilon XOP\epsilon Limit$.

The closed form expression is a throughput model and as such is based on the inequality $\lambda_a < \mu_a$, which guarantees the queue stability. Thus solving the equation $\lambda_a = \mu_a$ for $N\epsilon$based on the packet arrival rate at the AP (Eq. (3.12)) and the packet service rate (Eq. (3.20)), the VoIP call capacity is then derived at the point when the packet arrival rate (λ_a) exceeds the packet service rate (μ_a) at the AP. Therefore, the closed form expression for VoIP capacity in an IEEE

802.11 infrastructure WLAN as a function of η, denoted as $f(\eta)$, is given as

$$f(\eta) = \frac{1}{2}\sqrt{\left(\frac{\alpha + (\eta - 1)\,T_s^*}{\beta}\right)^2 + \frac{4\,\eta}{\beta}} - \frac{\alpha + (\eta - 1)\,T_s^*}{2\beta}, \tag{4.1}$$

where

$$\alpha = \left(T_s + (\overline{w}_a - /_a + 1 - c_a)\sigma + c_a T_{AIFS} + \frac{\bar{t}_a}{2}\right)\lambda_n,$$

$$\beta = \left(\frac{\lambda_n}{\mu_a} T_s + \frac{\lambda_n \bar{t}_n}{2\mu_a}\right)\lambda_n.$$

Note that all parameters above are given as in Section 3.4.

4.6.1 Analysis and Results

To study the accuracy of the $M/G/1/\infty$ model, we compare the maximum number of calls calculated using Eq. (4.1) with results obtained from Eq. (3.21) which is based on the $M/G/1/K$ model. Results for G.729 and G.711 codecs with 10 ms and 20 ms sampling rates based on both models are depicted in Figs. 4.8 and 4.9. It can be seen that the results show good agreement over a range of TXOP values. These results show that the voice capacity in an IEEE 802.11 infrastructure WLAN can be calculated using either $M/G/1/K$ or $M/G/1/\infty$ models. However, note that for the latter we do have a closed form expression for the maximum number of calls.

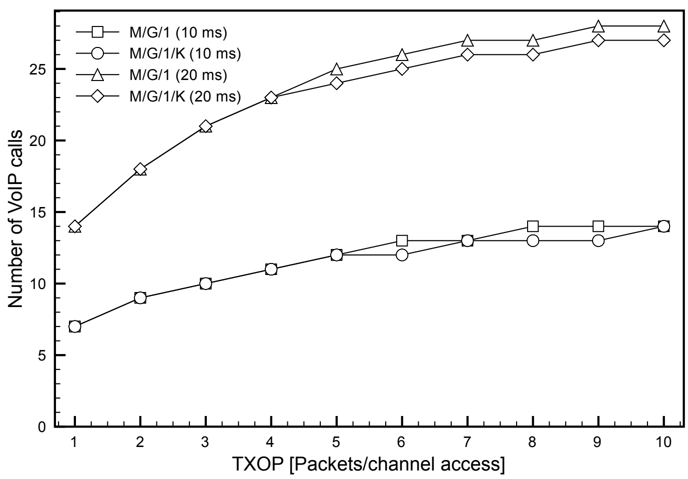

Figure 4.8: Comparison of the maximum number of G.729 VoIP calls with a 10 ms and 20 ms sampling rate for the two different analytical queueing models

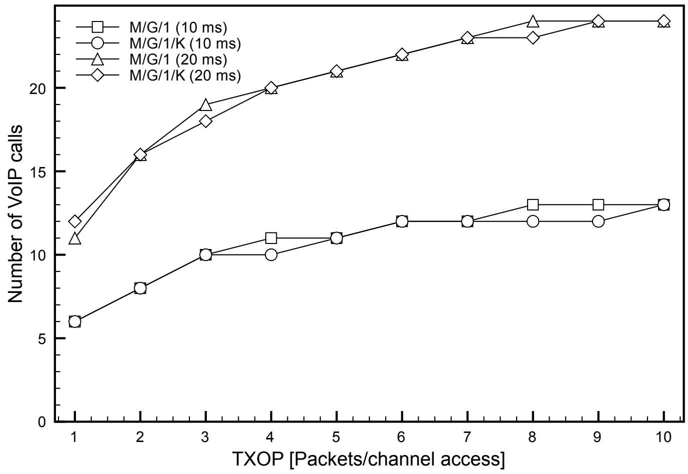

Figure 4.9: Comparison of the maximum number of G.711 VoIP calls with a 10 ms and 20 ms sampling rate for the two different analytical queueing models

Variable bit rate voice streams

Given that the buffer requirements at the AP are the same for variable bit rate voice streams, as shown in section 4.5, the closed form expression of the $M/G/1/\infty$ model can be used to obtain the voice capacity by adjusting the packet arrival rate as discussed in section 3.6.6. In Fig. 4.10 we show the number of G.729 VBR voice calls that can be maintained for a wide range of TXOP values, obtained used both, the $M/G/1/K$ and the $M/G/1/\infty$ model. It can be seen that the results are in good agreement. This confirms that either model can be used to derive the voice capacity in wireless networks for variable bit rate voice streams.

4.7 VoIP Capacity Approximation

In this section we propose a simple yet accurate approximation to estimate the number of voice calls in an IEEE 802.11 WLAN. To this end, we provide a heuristic recursive formula which provides the maximum number of voice calls for given TXOP parameter based on the previously obtained capacity. The approximation formula is a simple alternative to estimating the voice capacity in a WLAN, as it does not require the repeated calculation of the fixed-point formulation developed in Section 3.4. This formula also allows us to gain further insight into the voice capacity as a function of the TXOP parameter. In particular, we show that the voice capacity only depends on the TXOP parameter and C_1. Furthermore, we use the approximation to obtain the optimal value of the

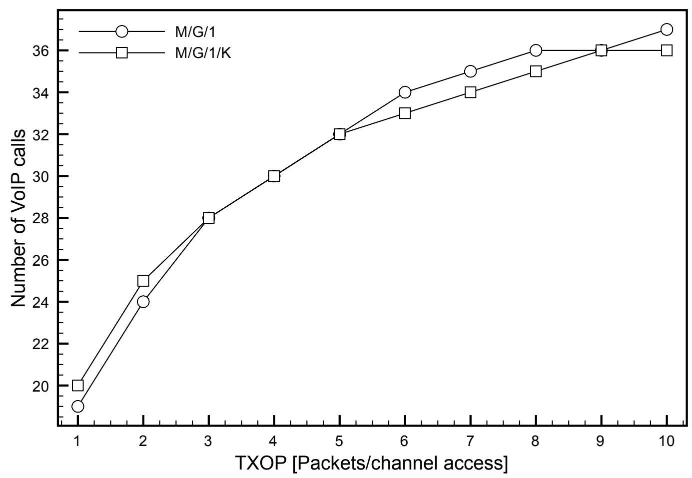

Figure 4.10: Comparison of the maximum number of variable bit rate G.729 VoIP calls with a 10 ms sampling rate for the two different queueing models, obtained analytically

TXOP parameter that maximizes the voice capacity. We argue that this optimal value is also a threshold value where the AP is no longer the bottleneck limiting the number of voice calls that the WLAN can support.

Voice capacity approximations have been used before to obtain the number of voice calls a WLAN can support. Hole and Tobagi [73], for example, derive the number of voice calls using (Eq. 2 in [73])

$$C\epsilon= \frac{1}{(2T_s+\overline{w})\lambda\epsilon}.\epsilon$$

Note that the notation above has been adjusted to be in line with the notation used in this work. Even though the above approximation provides reasonable accurate results, it does not consider collisions in the medium and also can not compensate for variable parameter changes such as the QoS MAC parameter, i.e. $T\epsilon XOP\epsilon Limit$. Furthermore, the equation does not allow a further investigation to identify the reason behind the limited VoIP capacity.

Our novel approach is different to the approximation in [73], because our approximation is more versatile and it captures a variety of parameters that are not considered in [73].

In Chapter 3 and 4 we have shown that for $\eta\epsilon= 1$ the WLAN can accommodate C_1 duplex calls and that the AP is the bottleneck of the network. In other words, there are, on average, C_1 packets sent in each direction of the bidirectional traffic flow in the network. Increasing $\eta\epsilon$from 1 to 2 enables the AP to send C_1 packets in $C_1/2$ channel accesses (i.e., 2 packets per each channel access) and thus has, on average, another $C_1/2$ channel accesses left on the downlink. Thus in addition to the C_1 voice calls that have already been supported previously, the AP can send an additional $C_1/2$ voice packets on the downlink.

As long as the AP is the bottleneck in the WLAN, the number of additional calls in this scenario (i.e. $\eta\epsilon = 2$) can be approximated as $\frac{C_1}{2}$ $/2$. This is because each additional call adds an equal number of packets to the uplink and the downlink voice traffic. As a result, the total number of calls using $\eta\epsilon = 2$ is $C_1 + \left\lceil \frac{C_1}{2} \;/2 \right\rceil$ The above approach is illustrated in Fig. 4.7 where C_1 is assumed to be four calls. In this scenario, increasing $\eta\epsilon$ from 1 to 2 reduces the number of packets sent by the AP by 50% which then enables us to add a new call as depicted in Fig. 4.7.

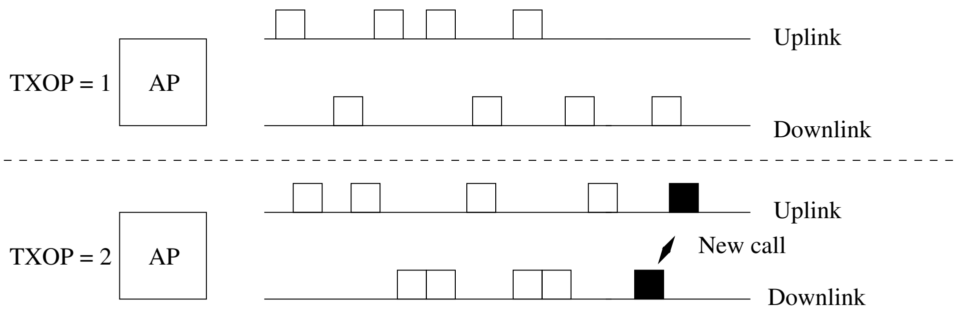

Figure 4.11: Illustration of the concept used to develop the voice capacity approximation.

Based on similar arguments the voice capacity with increasing TXOP value can be approximated as follows:

$$
\begin{aligned}
\eta\epsilon &= 1 \rightarrow C_1, \epsilon \\
\eta\epsilon &= 2 \rightarrow C_1 + \left\lceil \left(\frac{C_1}{2} \;/2 \right\rceil \right. \\
\eta\epsilon &= 3 \rightarrow C_1 + \left\lceil \left(\frac{C_1}{2} \;/2 \right\rceil + \left\lceil \left(\frac{C_1}{3} \;/2 \right\rceil \right. \right. \\
&\vdots \\
\eta\epsilon &= n\epsilon \rightarrow C_1 + \left\lceil \left(\frac{C_1}{2} \;/2 \right\rceil + \ldots + \left\lceil \frac{C_1}{\eta\epsilon} \;/2 \right\rceil \right. \\
&= C_1 + \sum_{\eta=2}^{n} \left\lceil \frac{C_1}{\eta\epsilon} \;/2 \right\rceil
\end{aligned}
\tag{4.2}
$$

Based on Eq. (4.2) and the closed form expression in Eq. (4.1)) for voice capacity with default TXOP value ($\eta\epsilon = 1$), the number of additional voice calls for arbitrary TXOP setting ($\eta > 1$), denoted as Γ_η, can be approximated by

$$\Gamma_\eta = \frac{f(\eta = 1)}{\eta} / 2. \tag{4.3}$$

We can now define a recursive formula for the obtained voice capacity. Let $\tilde{f}(\eta)$ denote the estimated number of VoIP calls for a given TXOP value. Then the VoIP capacity approximation is given by

$$\tilde{f}(\eta) = \begin{cases} \Gamma_1, & \eta = 1, \\ \tilde{f}(\eta - 1) + \Gamma_\eta, & \eta > 1. \end{cases} \tag{4.4}$$

In Figs. 4.12 and 4.13 we compare approximation results with results obtained by the analytical model for the G.729 and G.711 codecs using a variety of parameters. Observe that the approximation results match those of the analytical model closely even with large TXOP values (e.g. $\eta = 10$) despite the fact that the approximation may be too optimistic at those values. It is optimistic because the approximation is based on the argument that new calls can be initiated by increasing the TXOP value at the AP as long as the average rate of successful transmissions on the channel remains the same. This argument, however, ignores the fact that some transmissions (i.e. those originated from the AP) now involve multiple packets and therefore capture the channel for much longer. On the other hand, further increasing η beyond the above value carries no benefit because of the bottleneck shift as discussed in the previous section. Thus the results in Figs. 4.12 and 4.13 show that the approximation is reasonably accurate in terms of the voice capacity for the range of the TXOP values that are of interest. The approximation formula provides a simple alternative to the complete analytical model that can be used to estimate the voice capacity attainable in WLAN.

The approximation formula in Eq. (4.4) also allows us to gain further insights into the voice capacity in WLAN and to study its relation to MAC parameters such as the TXOP parameter. In particular, from Eqs. (4.3) and (4.4) it can be seen that the voice capacity depends on the TXOP value and the initial voice capacity obtained using default parameter settings (i.e. TXOP = 1). Also Γ_η in Eq. (4.4) tends to zero with an increasing value of TXOP parameter and thus $\tilde{f}(\eta) \approx \tilde{f}(\eta - 1)$ as TXOP $\to \infty$. This implies that the asymptotic value described in Section 3.6 for the maximum number of voice calls still exists even though the AP's buffer is now infinite. Furthermore, it can be observed

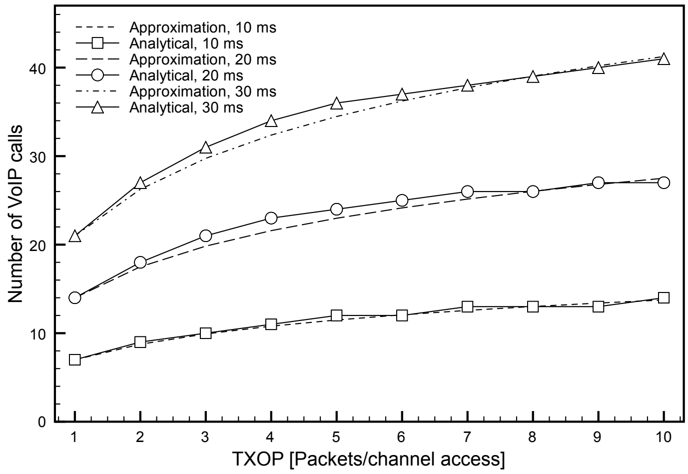

Figure 4.12: Comparison of the number of G.729 voice calls obtained by Eqs. (4.1) and (4.4) for different sampling rates and increasing TXOP parameter

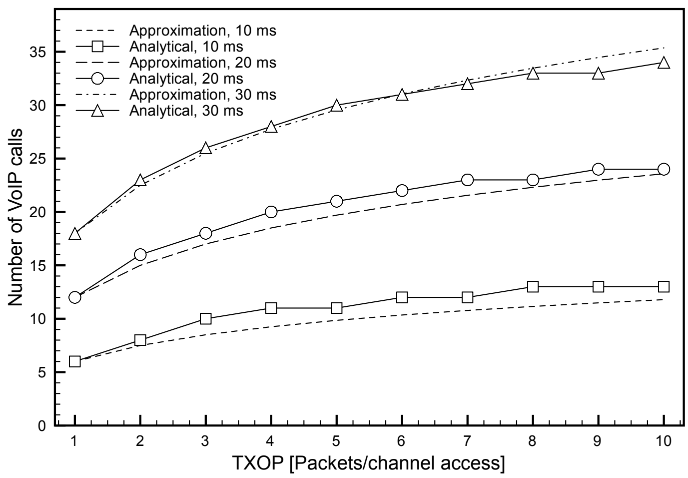

Figure 4.13: Comparison of the number of G.711 voice calls obtained by Eqs. (4.1) and (4.4) for different sampling rates and increasing TXOP parameter

in Eq. (4.2) that no additional packets can be gained on the downlink as TXOP value exceeds C_1 indicating that the AP buffer is no longer a bottleneck as described in Chapter 3.4.

4.7.1 Variable bit rate voice flows

In Fig. 4.14 we show the number of variable bit rate G.729 voice calls with a 10 ms and 20 ms sampling rate that can be supported in the WLAN, obtained analytically and using the VoIP capacity approximation. The results show that in this scenario the VoIP capacity approximation can also be used for variable bit rate voice flows, and shows the versatility of our novel approach.

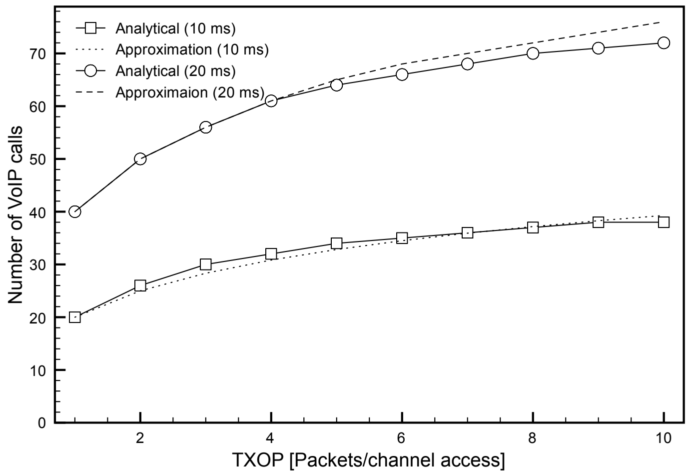

Figure 4.14: Comparison of the number of variable bit rate G.729 voice calls obtained by Eqs. (4.1) and (4.4) for different sampling rates and increasing TXOP parameter

4.8 Summary

In this Chapter we studied the impact the buffer size has on the VoIP capacity in an IEEE 802.11 infrastructure WLAN. Using our analytical model developed in Chapter 3 and simulation we showed that there is a minimum buffer size (K_{min}) with which the number of voice calls that can be supported with an acceptable level of quality is maximum, and that increasing the buffer size will not result in an increased voice capacity. In particular we showed that a buffer size of $K\epsilon = 30$ packets is sufficient to attain the maximum number of VoIP calls an IEEE 802.11 infrastructure WLAN can support. We showed that this minimum buffer size is consistent for a wide range of scenarios, including different voice codecs, scenarios where a channel access preference is given to the AP using the different adjustable MAC parameters as well as variable bit rate voice streams.

Based on our finding that the voice capacity in a WLAN is independent of the buffer size, given a minimum buffer size, we developed an analytical model based on the $M/G/1/\infty$ queueing model. Specifically, we developed a closed-form expression for the number of voice calls a WLAN can support where there is infinite buffer space as well as preference is given to the AP using the $TXOPLimit$ parameter as highlighted in Chapter 3. Using a wide range of parameters we confirmed that the voice capacity can be obtained using either an

$M/G/1/K$ or an $M/G/1/\infty$ queueing model. We had shown previously that the buffer requirements for variable bit rate voice streams are equal to constant bit rate voice streams. Similarly we showed that the closed-form expressions can also be used for VBR voice flows.

Using our closed-form expression for the voice capacity in an IEEE 802.11 infrastructure wireless network we have proposed a novel approach to determine the voice capacity when preference is given to the AP using the adjustable $TXOPLimit$ parameter η. Our approach is based on a recursive equation based on the additional channel access gained when the AP is allowed to transmit multiple packets during a single channel access period. Using the approximation formula, we showed that the voice capacity in a WLAN only depends on the initial voice capacity when there is no preference given to the AP, and the number of packets the AP can transmit per channel access. Furthermore, we obtained analytically the asymptotic voice capacity outlined in Chapter 3. Based on our findings we could also confirm our claim that our proposed ideal value of the TXOP parameter, $\eta = C_1$ is optimal. Finally, we confirmed that the approximation can also be used for variable bit rate voice streams, thus showing the versatility of this novel approach.

5

Dynamic codec with priority for Voice over IP in IEEE 802.11 WLAN

In the previous chapters we provided an analysis of the VoIP capacity in IEEE 802.11 infrastructure WLANs and showed that the number of VoIP calls that can be supported with an acceptable level of quality is surprisingly low. Subsequently we evaluated a solution based on the adjustable medium access control parameter and showed that a substantial voice capacity gain can be achieved when priority is assigned to the access point. We investigate this using an analytical model, simulation and testbed measurements.

In this Chapter, we propose a novel scheme utilizing the above channel access priority in conjunction with dynamic voice codecs. A recent development in VoIP is the use of *dynamic* voice codecs which are designed to adapt to changes in network conditions. For example, SILK [58, 160] used in Skype V.4 or SPEEX [161] used in Google Talk, monitor the call quality and adjust the codec parameters accordingly. The goal is to maintain the call with reduced quality during the congestion period of the network. In this context, mixed codecs can be seen in some of the previous work. In particular, the authors of [26] consider two types of voice traffic using different codecs. In their work however, it is assumed that both types of voice traffic share the channel and are served by the AP without traffic differentiation. Also in [162] a dynamic adaptation of the voice codec is proposed to suit the change in transmission rate of the WLAN.

Our proposed scheme is based on the IEEE 802.11 quality of service (QoS) mechanism to exploit the tradeoff between the codec quality and priority to maintain a call during periods of high contention without compromising the individual call quality. At the same time the proposed scheme also increases the overall number of calls that can be supported by the WLAN. The novelty in our approach is that our scheme gives incentive to users who are willing to use a lower quality codec and thus reduce the overall contention when there is a high traffic load in the medium. The incentive is implemented by way of giving priority at the AP to traffic originating from lower quality codec users. Priority is assigned using a smaller CW_{min} parameter at the AP rather than the $TXOPLimit$ parameter used in Chapter 3. Using the contention window rather than the $TXOPLimit$, ensures that VoIP calls using the lower quality voice codec gain a guaranteed channel access advantage over the other high quality voice calls. We will explore the option of different $TXOPLimit$ parameter for the traffic prioritization later on in this chapter. This scheme can be easily implemented because the priority is only applied at the AP where the network bottleneck is located. Also, users can voluntarily choose the codec by monitoring their own call quality as is done in Skype.

In order to understand the benefits of the above proposed scheme, we develop a detailed analytical model based on our previous work in Chapter 3. In particular, we extend the existing model to accommodate multi-codec voice streams and to include internal collisions caused by the use of priority at the AP. Note that in [120] and [145] the authors investigated the use of multiple queues based on the IEEE 802.11e protocol, however, they did not consider the effect of internal collisions. We will show through our analytical model that the impact of this internal collision can not be neglected.

Furthermore we assess the quality of calls in the WLAN using the proposed scheme using the ITU-T E-model [19] as discussed in Chapter 2. Note that although the analytical model has been developed to analyze the scenario outlined in Section 5.1, the proposed model is sufficiently versatile to enable study of a range of scenarios, i.e. the interactions between voice and video traffic in a WLAN using the IEEE 802.11 QoS mechanism. Additionally, the model allows the investigation of a range of network parameters such as collision probability and queue utilization at a station and different network metrics, such as throughput, can be derived. The model is also flexible enough to be extended to incorporate other traffic types, such as traffic generated by TCP connections.

To this end, our main contributions can be summarized as follows.

1. We propose a novel scheme to reduce the overall contention in a highly congested WLAN, based on dynamic voice codecs and channel access priorities.

2. We show that the proposed scheme improves the voice capacity while maintaining an acceptable quality for all individual calls, evidenced by results obtained from the ITU-T E-model.

3. We develop a detailed analytical model to obtain the voice capacity in a multi-codec environment that also takes into account the AP's internal collision, to show the benefits of our proposed scheme and that the voice capacity can be significantly improved if channel access priority is used in conjunction with dynamic voice codecs.

4. We show that internal collisions can have a significant impact on the system performance.

5. We highlight that simply applying a dynamic codec without traffic prioritization will not be sufficient to significantly improve the voice capacity without compromising the call quality of all calls in the system.

The rest of the chapter is organized as follows. In Section 5.1 we describe in detail the proposed scheme with dynamic codecs and priority at the AP, followed by a description of the internal collision management of the IEEE 802.11 QoS mechanism in Section 5.2. The analytical model is then developed in Section 5.3. We validate our analytical results by simulation in Section 5.4, and conclude in Section 5.5.

5.1 Dynamic codec with priority scheme

Consider a scenario where multiple voice calls are initiated simultaneously in an IEEE 802.11e infrastructure wireless LAN, as depicted in Fig. 5.1.

As discussed in Section 2.2.3, the IEEE 802.11 protocol can provide QoS in WLANs using four access categories, each with different and adjustable

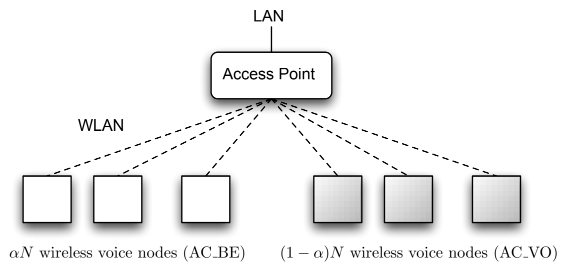

Figure 5.1: WLAN topology

MAC parameter settings as defined in the enhanced distributed channel access (EDCA) mechanism [14]. Recall that the first two access categories, *best effort* (AC_BE) and *background* (AC BK) can be understood as *data* categories, whereas *video* (AC VI) and *voice* (V_d) are specifically designed for real-time multimedia traffic. In this chapter, we propose the use of dynamic voice codecs where users can choose to switch to a lower quality codec to maintain their calls in a congested WLAN. To this end, both the sampling rate and the payload of the packet can be adjusted to form a low quality codec. Similar to existing VoIP implementation, such as Skype or Google Talk, users can monitor their own call quality and make the switching decision when appropriate. However, this action can lead to a lower perceived voice quality as sometimes observed in existing VoIP applications. To compensate for the reduction in voice quality and to give incentive to users who are willing to switch to a lower quality codec, higher priority is then given at the AP to traffic originated from those users. Thus placing traffic in the higher priority queue at the AP will encourage a less aggressive behavior from users in a highly congested medium. As a result not only can users maintain their call at an acceptable quality level using a lower quality codec, but the overall number of calls can also be increased.

The concept of our proposed scheme is depicted in Fig. 5.2. As shown, the user perceived voice quality was "good" initially. However, at some stage the call enters a *call deterioration phase*. This is caused by an increased level of contention in the WLAN, e.g. the network becomes saturated due to an increasing number of VoIP calls. If the user does not use our proposed scheme, as shown, the voice quality will drop and would be considered "bad". However, if the user uses our proposed scheme, the actual voice quality will drop, but the overall quality will still be "good", whereas all other calls, who do not adjust may experience a "bad" voice call quality.

Our proposed dynamic voice codec with priority (DCwP) is implemented as follows. Every voice user will contend for a channel access using the MAC

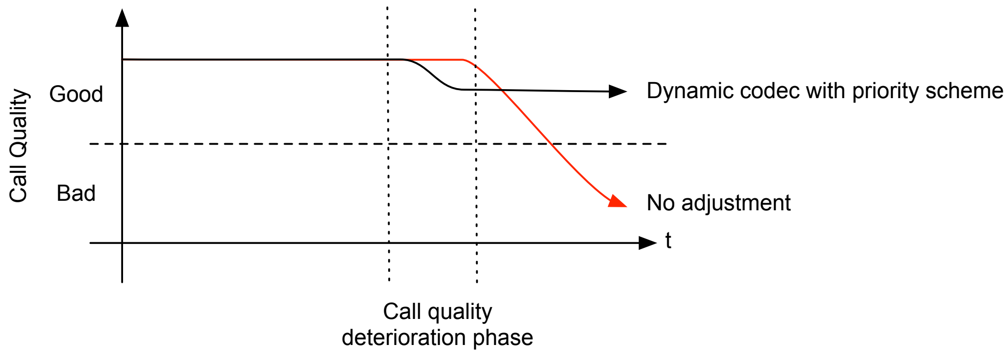

Figure 5.2: Dynamic codec with priority (DCwP) benefits for users who adopt the proposed solution or not

parameters defined in the AC BE category. Users will continue to use the same MAC parameters independent of the codec used. At the AP, however, different traffic originated from users equipped with different quality codecs will be placed into different access categories. In particular, traffic from users with high quality codecs will be assigned to the AC BE category, while other voice traffic will be mapped into the AC VO category. Note that only contention windows (CW_{min}, CW_{max}) are used in these two categories to differentiate different traffic at the AP. Although other MAC parameters such as $TXOPLimit$ or the arbitrary interframe space ($AIFS$) can also be used, the use of the contention window is the simplest way to prioritize traffic and to demonstrate the benefit of the scheme proposed in this chapter.

Herein we consider the voice capacity to be the maximum (total) number of voice calls a WLAN can support with an adequate level of quality. As in Chapter 3 we define κ as a packet loss threshold, beyond which a user-perceived quality of a call can no longer be maintained, and is set to 2%. Additionally, we will also consider an end-to-end delay bound of 60 ms [27]. Even though the ITU-T G.114 specification [185] allows a 150 ms end-to-end delay before the call quality is no longer considered to be good, this delay however, is the total delay from the sender to the receiver. In this work, we do not consider the path outside the WLAN, and as such, the experienced delay in the one-hop infrastructure WLAN should be well below 150 ms. We make use of the ITU-T E-model [19] which uses delay and loss to obtain a measure (the R-value) that estimates the user-perceived voice quality. To avoid ambiguity, we denote the R-value of the ITU E-model with R_Q in this chapter and use R as the retry limit defined by the backoff process of the IEEE 802.11 medium access control.

5.2 Internal Collision Management

Because the IEEE 802.11 QoS mechanism defines the different access categories with different priorities an internal collision can occur. An internal collision occurs if the backoff process of at least two access categories reaches zero at the same time. To resolve an internal collision, the protocol uses an internal collision resolution mechanism, whereby the packet of the higher access category gains channel access, and the packet of the lower priority access category is re-scheduled for transmission after an additional backoff. In Fig. 5.3 we show a stylized internal collision. As shown the AC_BE and the AC_VO queue transmit the packet and an internal collision occurs (indicated by *flash* symbol). As shown the AC_VO packet is then passed to the channel, whereas the AC_BE packet is scheduled for retransmission. Due to the internal collision resolution mechanism an AC_VO packet will always receive channel access, and thus can only collide with other packets in the wireless medium. Packets from all other
the wireless ch

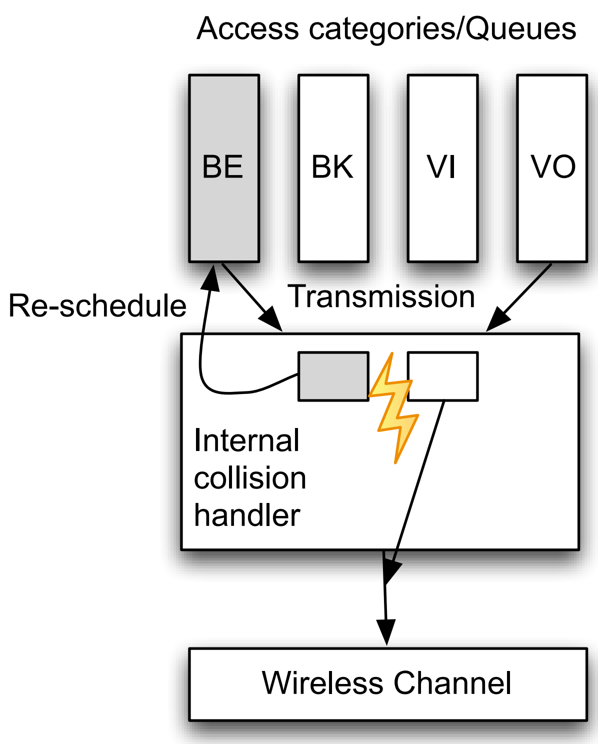

Figure 5.3: Internal collision at the AP

Note that whenever a packet is re-scheduled for retransmission, the packet is placed at the head of line (HOL) of that queue. Also note that if a packet is re-scheduled, its retry counter is increased. Therefore, a packet could reach the retransmission limit due to internal collisions only, and thus can be discarded.

5.3 Modeling approach

In our analysis we consider an IEEE 802.11 infrastructure WLAN, consisting of one AP and N wireless nodes, as depicted in Fig. 5.1.

The AP has two access categories V_d and V_p, and the two access categories carry only voice flows using a specific codec, i.e. G.711 with a 10 ms sampling rate in V_d and G.729 with a 20 ms sampling rate in V_p, respectively. We refer to wireless nodes served by the V_d queue of the AP as *standard* wireless nodes and as *priority* wireless nodes whenever they are served by the V_p queue. Note that the V_d and the V_p queue are equivalent to the AC_BE and the AC_VO access category of the IEEE 802.11 QoS mechanism. In this WLAN there are αN, $0 < \alpha < 1$ standard wireless nodes and $(1 - \alpha)N$ priority wireless nodes. Each wireless node maintains a full-duplex voice call to a node outside the WLAN. In this scenario we assume EDCA basic access is used over an ideal channel without interference or hidden terminals. Unless otherwise stated, the indices a and n correspond to the AP and a wireless node, respectively. Also, to differentiate between the two access categories, the priority access category (V_p) is tagged using an asterisk (*).

Let λ_n and λ_{n^*} denote the packet arrival rate at a standard and priority wireless node, respectively. The total average packet arrival rate at the AP is given by $\lambda_A = \alpha N \lambda_n + (1 - \alpha) N \lambda_n^*$, and the packet service rate is denoted by μ for each *station* (wireless node and/or AP). Similar to the analytical model in Section 3.4 here we assume that the traffic flows arrive at a node according to a Poisson process, the AP and a wireless node can then be modeled as an $M/G/1/K$ queue, where K is the number of packets that can be queued at a station and can take different values for the AP and for the wireless nodes. Furthermore, the queue utilization is defined as $\rho = \lambda/\mu$, where ρ is also the probability that a station has a packet to send [176]. Consequently, a station will be idle with probability $1 - \rho$.

Our analysis is based on a fixed-point formulation between the collision probability seen by a packet submitted to the channel, and the condition attempt probability per slot. Due to the variability of the collision probability, the average packet service time is different for the AP and the wireless node, and is further influenced by the used access category at the AP and the traffic type. Therefore, we first discuss the generic fixed-point, before we describe the individual collision probabilities and the average packet service time for the AP, the standard and the priority wireless voice nodes.

5.3.1 Fixed-point formulation

As our analysis is built around the fixed-point formulation between the collision probability c_i and the attempt probability τ_i of a station, $i \in \{a, n, a^*, n^*, A\}$, we define the fixed-point equations as follows:

$$c_i = f(\rho_i, \tau_i), \tag{5.1}$$

$$\tau_i = g(c_i, w_i). \tag{5.2}$$

Recall that ρ_i in Eq. (5.1) is the probability that a station has a packet to send and depends on the packet arrival rate (λ_i) and the packet service time ($1/\mu_i$) as defined previously. Furthermore, τ_i in Eq. (5.1) is given in Eq. (5.2) and can be calculated using

$$\tau_i = \frac{\sum_{j=0}^{R} c_i^j}{w_i},$$

where w_i in Eq. (5.2) is the average backoff experienced by a station and can be obtained similarly to Eq. (3.9) using

$$w_i = \sum_{j=0}^{R-2} (1 - c_i) c_i^j \frac{\sum_{k=1}^{j} \phi_k W - 1}{2} + c_i^{R-1} \frac{2^m W - 1}{2}. \tag{5.3}$$

Note that R is the maximum retry limit as defined in the IEEE 802.11 protocol [4], W is the current contention window and $\phi_k = 2$ for $j \leq m$ and 1 otherwise, as in Section 3.4.

To solve Eq. (5.1) requires the average packet service time. As the service time is variable depending on the station and the type of traffic sent, we will discuss the service time and derive the collision probability c_i in individual sections.

5.3.2 Access Point

We now apply the technique discussed in Section 5.3.1 to obtain the average service rate of the AP. As outlined in Section 5.1, our scheme is implemented by way of using two different access categories at the AP, similar to those of the IEEE 802.11 QoS mechanism [4]. Because the IEEE 802.11 protocol defines different access categories with different priorities, the collision probability of the AP depends on two parts: i) the internal collision probability, and ii) the external collision probability. As the AP has multiple traffic categories, an internal collision can occur if the backoff process in both access categories reaches zero simultaneously. The IEEE 802.11 protocol defines an internal collision resolution mechanism to handle these collisions. In case of an internal collision, the internal collision handler will grant access to the wireless medium to the higher access category, here V_p. The packet of the lower access category (V_s) is rescheduled for transmission after an additional backoff. If no internal collision occurs, the packet can still collide on the wireless medium (referred to as an external collision). If an external collision occurs, the packet is rescheduled for transmission in its respective access category, as outlined above.

Let $\delta\epsilon$denote the internal collision probability at the AP seen by a non-priority (standard) packet. Recall that an internal collision between two access categories at the AP only occurs if the random backoff process in both access categories reaches zero in the same instance. Then conditioned on the fact that the AP has a standard packet to be sent, the internal collision probability seen by a standard VoIP packet is then defined as

$$\delta\epsilon = 1 - \left(1 - \frac{\lambda_{n^*}}{\mu_A}\tau_{a^*}\right),\epsilon \tag{5.4}$$

where $(\lambda_{n^*}/\mu_A)\tau_{a^*}$ is the conditional attempt probability of the priority access category. Therefore, with probability $1 - \delta\epsilon$no internal collision occurs.

If no internal collision occurs, the packet can still collide externally with packets transmitted by either, a standard or a priority wireless node. The external collision probability for packets sent by the AP of either access categories is equal. This is because both packet types can only collide with packets transmitted by both a standard or a priority wireless voice node. Therefore, based on Eq. (5.1), the collision probability of a priority packet (which is equal to the external collision probability of any packet) is given by

$$c_{a^*} = 1 - \left((1 - \rho_n\tau_n)^{\alpha N}(1 - \rho_{n^*}\tau_{n^*})^{(1-\alpha)N}\right).\epsilon \tag{5.5}$$

Note that similar to the case of τ_{a^*}, τ_n and τ_{n^*} can be calculated using Eq. (5.2) using the respective collision probability and backoff.

The collision probability of a standard packet sent by the AP c_a can be derived based on Eqs. (5.5) and (5.4) and the probability that a standard packet will be transmitted successfully, given by $(1-\delta)(1-c_{a^*})$, or be discarded if the retry limit R has been reached. Thus for $R \to \infty$, we can establish a recursive formula to obtain the collision probability c_a, which is given by

$$
\begin{aligned}
R &= 1 \to c_{a,1} = \delta + (1-\delta)c_{a^*} \\
R &= 2 \to c_{a,2} = \delta c_{a,1} + (1-\delta)c_{a^*}c_{a,1} \\
R &= 3 \to c_{a,3} = \delta c_{a,2} + (1-\delta)c_{a^*}c_{a,2} \\
&\vdots \\
R &= n \to c_{a,n} = \delta c_{a,n-1} + (1-\delta)c_{a^*}c_{a,n-1},
\end{aligned}
$$

and therefore, for a maximum retry limit R $(R > 1)$, the collision probability of a standard packet sent by the AP is given as

$$c_a = \delta c_{a,i-1} + (1-\delta)c_{a^*}c_{a,i-1},\ i = 2,..,R. \tag{5.6}$$

The above recurs

5.5.

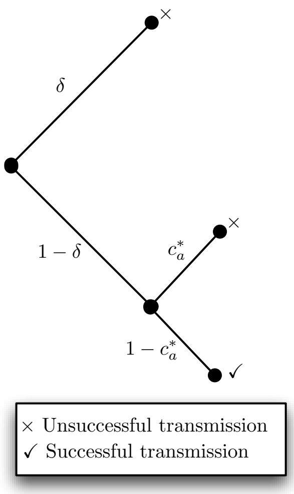

Figure 5.4: Collision probability c_a for $R = 1$

As shown in section 5.3.1, the fixed-point equation given in Eq. (5.1) depends on the attempt probability τ_i and the probability that a station i has a packet to send (ρ_i), which in turn is defined by the packet arrival rate λ_i and the

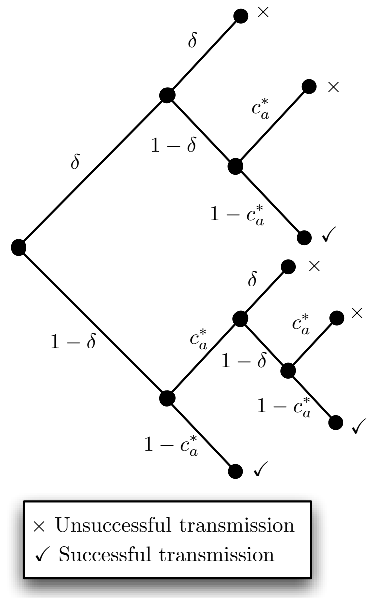

Figure 5.5: Collision probability c_a for $R = 2$

packet service rate μ_i of that station. Following our previous approach in [186], the average service time can be decomposed into three parts: i) the successful transmission and collision of the packet itself, ii) the successful transmissions and collisions of other stations, and iii) the mandatory (average) backoff of a station.

Equivalent to the analytical model in Chapter 3, for the first part, the successful transmission time of a packet is given as

$$T_s = T_{AIFS} + T_p + T_{SIFS} + T_{ACK}, \epsilon \tag{5.7}$$

$$T_{s^*} = T_{AIFS} + T_{p^*} + T_{SIFS} + T_{ACK}, \epsilon \tag{5.8}$$

and the collision time is given by

$$T_c = T_{AIFS} + T_p + T_{ACK_{TO}}, \epsilon \tag{5.9}$$

$$T_{c^*} = T_{AIFS} + T_{p^*} + T_{ACK_{TO}}. \epsilon \tag{5.10}$$

Note that T_p, T_{p^*}, T_{ACK} and $T_{ACK_{TO}}$ are the transmission times of a standard packet, a priority packet, the acknowledgement and $T_{ACK_{TO}}$ is the ACK timeout period of an unsuccessful transmission. Also note that T_p and T_{p^*} can be different, depending on the voice codec and its sampling rate. Additionally, T_{AIFS} and T_{SIFS} are the duration of the arbitrary interframe space (AIFS) and the short interframe space (SIFS) in microseconds, respectively, as defined in the IEEE 802.11 protocol.

As a packet can collide multiple times before its successful reception, the average collision time depends on the packet itself, e.g. a standard or a priority packet, and whether it collides with a standard or a priority packet. Therefore the average collision time of a standard and a priority packet transmitted by a station can be calculated by

$$\frac{t_i}{2} \approx \begin{cases} \frac{\sum \gamma T_c c_i + (1-\gamma) T c^* c_i}{2(1-c_i)}, & i \in \{n, a\} \\ \frac{\sum T c^* c_i}{2(1-c_i)}, & i \in \{n^*, a^*\} \end{cases} \tag{5.11}$$

where γ is the probability of a standard packet begin transmitted by either the AP or a standard wireless voice node, which is given as follows

$$\gamma = \frac{\alpha N \lambda_n}{\alpha N \lambda_n + (1-\alpha) N \lambda_{n^*}}, \tag{5.12}$$

Note that the 1/2 factor in Eq. (5.11) is based on the assumption that a collision is due to simultaneous transmission from two stations only, and thus the average collision time caused by a packet is half of the total collision time experienced by all stations.

Now Eq. (5.11) can be used to derive the average packet collision time for both type of packets transmitted by the AP.

The second component of the average packet service time of the AP is the interruption to the backoff process due to successful transmission and collisions of packets by other stations. Note that the interruption of the backoff of the AP also includes the interruption caused by an internal collision conditioned such that a standard packet collides with a priority packet as given bt

$$\delta \quad (1-\alpha) N \frac{\lambda_{n^*}}{\mu_A} \quad T_{s^*} + \frac{t_{a^*}}{2} \quad \sigma \tag{5.13}$$

Furthermore, for the AP, the interruption to the backoff due to successful transmission and collisions by the wireless nodes depends on whether the AP attempts to transmit a standard or a priority packet. However, independent of the traffic type, a packet sent by the AP can collide with packets sent by the αN standard wireless voice nodes and packets sent by the $(1-\alpha)N$ priority wireless voice nodes. Hence, the total interruption to the backoff of the AP is given by

$$
\begin{aligned}
&(1-\not\delta)\gamma N\epsilon\ \alpha\epsilon\frac{\lambda_n}{\mu_A}\quad T_s+\frac{t_n}{2}\\
&+(1-\not\alpha)\frac{\lambda_{n^*}}{\mu_A}\quad T_{s^*}+\frac{t_{n^*}}{2}\qquad\sigma\epsilon\\
&+(1-\not\gamma)N\quad\alpha\epsilon\frac{\lambda_n}{\mu_A}\quad T_s+\frac{t_{n^*}}{2}\\
&+(1-\not\alpha)\frac{\lambda_{n^*}}{\mu_a}\quad T_{s^*}+\frac{t_{n^*}}{2}\qquad\sigma.\epsilon
\end{aligned}
\tag{5.14}
$$

The third component for the average packet service time of the AP is the mandatory backoff before attempting to transmit a packet. However, for the AP the total average backoff of the AP is not simply w_i as in Eq. (5.3). This is because to obtain the total average backoff of the AP requires the average backoff of each access category, and the probability of transmitting a standard or a priority packet. Furthermore, as outlined in [89] and shown in Chapter 3, the backoff counter in EDCA is managed differently to DCF, and as Eq. (5.3) only captures the DCF process, compensation for the difference is required. Therefore, the total average backoff of the AP is calculated using

$$
\begin{aligned}
w_A = ((\gamma w_a + (1-\not\gamma)w_{a^*}) - /_a)\,\sigma\epsilon&\\
+\gamma((1-\not c_a)\sigma\epsilon + c_a T_{AIFS})&\\
+(1-\not\gamma)((1-\not c_{a^*})\sigma\epsilon + c_{a^*} T_{AIFS}),\epsilon&
\end{aligned}
\tag{5.15}
$$

where $_i, i\epsilon\in \{a, \epsilon n\}$ $\not c$an be derived based on the interruptions of the backoff process as given in Eq. (5.14).

Then, following our approach in Chapter 3, the total average packet service time of the AP can be calculated using

$$
\begin{aligned}
\frac{1}{\mu_A} = \gamma \left(T_s + \frac{t_a}{2} \right) &+ (1-\gamma)\left(T_{s^*} + \frac{t_{a^*}}{2} \right) \\
&+\delta \left((1-\alpha)N\frac{\lambda_n^*}{\mu_A}\left(T_{s^*} + \frac{t_{a^*}}{2} \right) \right) \\
&+(1-\delta)\gamma N \left(\alpha\frac{\lambda_n}{\mu_A}\left(T_s + \frac{t_n}{2} \right) \right. \\
&\left. +(1-\alpha)\frac{\lambda_{n^*}}{\mu_A}\left(T_{s^*} + \frac{t_{n^*}}{2} \right) \right) \\
&+(1-\delta)(1-\gamma)N \left(\alpha\frac{\lambda_n}{\mu_a}\left(T_s + \frac{t_{n^*}}{2} \right) \right. \\
&\left. +(1-\alpha)\frac{\lambda_{n^*}}{\mu_A}\left(T_{s^*} + \frac{t_{n^*}}{2} \right) \right) \\
&+(\overline{w}_A - {}_a)\sigma + \gamma((1-c_a)\sigma + c_a T_{AIFS}) \\
&+(1-\gamma)((1-c_{a^*})\sigma + c_{a^*} T_{AIFS}).
\end{aligned} \tag{5.16}
$$

Then using the packet arrival rate λ_A, the collision probabilities for each traffic type at the AP given in Eqs. (5.5) and (5.6) as well as the average packet service time in Eq. (5.16), the fixed-point equations in Eqs. (5.1) and (5.2) can be solved iteratively.

5.3.3 Standard wireless voice nodes

We now turn our attention to the standard wireless VoIP nodes transmitting the high quality voice codec VoIP traffic. To solve the fixed-point equations in Section 5.3.1 for the standard wireless voice nodes, the collision and attempt probability as well as the average packet service time is required. We follow the approach for the AP as presented in the previous section. Note that we assume that each wireless voice node only carries one type of traffic, and thus, does not experience any internal collisions. Hence, a wireless voice node attempting to transmit a packet can only collide with the remaining nodes in its traffic class, with all nodes of the other traffic class, as well as packets of either traffic class sent by the AP. Therefore the (external) collision probability of a standard wireless node is given as

$$
c_n = 1 - (1-\rho_n\tau_n)^{\alpha N - 1}(1-\rho_{n^*}\tau_{n^*})^{(1-\alpha)N}(1-\rho_A\tau_A), \tag{5.17}
$$

where $\rho_A \tau_A$ is the conditional attempt probability of the AP transmitting a packet in any slot, irrespective of its traffic class. Hence ρ_A is the probability that the AP has a packet to send and is defined by

$$\rho_A = \frac{\alpha N \lambda_n + (1-\not\alpha) N \lambda_{n^*}}{\mu_A} = \frac{\lambda_A}{\mu_A}, \epsilon \tag{5.18}$$

and τ_A is given as $\tau_A = \gamma \tau_a + (1-\not\gamma)\tau_{a^*}$.

Similar to the AP, the average packet service time of a standard wireless voice node $(1/\mu_n)$ can also be decomposed into the aforementioned three parts.

The first part, the successful transmission time of a standard packet is given in Eq. (5.7), and replacing c_i with c_n in Eq. (5.11) the average collision time for a standard packet can be obtained.

For the second component, the backoff process of a standard wireless voice node will be interrupted due to the successful transmission and collision of packets sent by the $\alpha(N\epsilon - \not 1)$ remaining standard wireless voice nodes, as well as the $(1-\not\alpha)N\epsilon$priority wireless voice nodes. Furthermore, successful transmissions and collisions of packets of both access categories sent by the AP, will further interrupt the backoff process of a standard wireless voice node.

The third component, the average backoff, can be calculated using Eq. (5.3) and appropriately replacing $c\epsilon$with c_n and calculating $\ _n$ based on the interruption to the backoff process (second component).

Following the argument for the AP, the average packet service time of a standard wireless voice node is then given by

$$\begin{aligned}
\frac{1}{\mu_n} &= \quad T_s + \frac{t_n}{2} \\
+(\alpha N\epsilon - \not 1)\rho_n &\quad T_s + \frac{t_n}{2} \\
+(1-\not\alpha)N\frac{\lambda_{n^*}}{\mu_n} &\quad T_{s^*} + \frac{t_{n^*}}{2} \\
+\frac{\lambda_A}{\mu_n} \quad \gamma \quad T_s + \frac{t_a}{2} &\quad + (1-\not\gamma) \quad T_{s^*} + \frac{t_{a^*}}{2} \\
+(w_n - /_n + (1-\not c_n))\sigma\epsilon &+ c_n T_{AIFS}.\epsilon
\end{aligned} \tag{5.19}$$

To solve Eqs. (5.1) and (5.2), we apply Eq. (5.17) to obtain the fixed-point for the standard wireless voice nodes.

5.3.4 Priority wireless voice nodes

The process to calculate the average service time of a priority wireless node is equivalent to that of a standard wireless voice node. However, it also requires the collision probability, transmission and collision times, the duration of the backoff, the interruptions to the backoff and the compensation for the different backoff in EDCA. The collision probability is similar to Eq. (5.17), because the packets transmitted by the priority wireless voice node can only collide with either packet types send by the AP, with packets send by the standard wireless voice nodes or with packets of the remaining priority wireless voice nodes. Hence, c_n^* can be calculated using

$$c_{n^*} = 1 - (1 - \rho_n \tau_n)^{\alpha N} (1 - \rho_n^* \tau_n^*)^{(1-\alpha)N-1} (1 - \rho_A \tau_A). \tag{5.20}$$

Following the approach for the standard wireless voice nodes, and replacing c_i with c_n^* where appropriate, the average packet service rate for the priority wireless voice nodes is then given by

$$\begin{aligned} \frac{1}{\mu_{n^*}} = {} & T_{s^*} + \frac{t_{n^*}}{2} \\ & + \alpha N \frac{\lambda_n}{\mu_{n^*}} \left(T_s + \frac{t_n}{2} \right) \\ & + (1-\alpha)(N-1) \frac{\lambda_{n^*}}{\mu_{n^*}} \left(T_{s^*} + \frac{t_{n^*}}{2} \right) \\ & + \frac{\lambda_A}{\mu_{n^*}} \left[\gamma \left(T_s + \frac{t_a}{2} \right) + (1-\gamma) \left(T_{s^*} + \frac{t_{a^*}}{2} \right) \right] \\ & + (w_{n^*} - w_n + (1 - c_{n^*}))\sigma + c_{n^*} T_{AIFS}. \end{aligned} \tag{5.21}$$

5.3.5 Obtaining the voice capacity in WLAN

To obtain the VoIP capacity in an IEEE 802.11 infrastructure WLAN with the DCwP scheme, we follow a similar approach used previously in Chapter 3. Solving the fixed-point given in Eqs. (5.2) and (5.1) we can derive the queue utilization ρ to determine the maximum number of VoIP calls ($\hat{C}$). Here we also require the packet loss (κ) at the AP to be less than 2% to have an acceptable level of call quality. Given that our analytical model is derived from the $M/G/1/K$ queueing model used in Chapter 3, here we also use the blocking probability of an $M/M/1/K$ queue [176] to approximate the packet loss at the

AP and thus can be calculated using

$$\kappa_A \approx \frac{(1-\rho_A)\rho_A^K}{1-\rho_A^{K+1}}, \epsilon \tag{5.22}$$

Note that the above approximation is acceptable to deduce the VoIP capacity in terms of number of voice calls, as discussed in Section 3.6.2. Also note that we validate this claim in the next section by comparing the number of VoIP calls that can be supported obtained analytically, by simulation and testbed measurements.

5.4 Analysis and Discussion

In this section we use our analytical model and ns-2 simulation to evaluate the performance of our proposed scheme. Furthermore, we also report on measurements obtained from experiments conducted in our IEEE 802.11 WLAN testbed to show the benefits of this scheme and to further validate our results obtained analytically and by simulation. The implementation of the two types of traffic is by way of using two different access categories at the AP, and a single access category for all wireless voice nodes. Note that the EDCA parameter for the wireless voice nodes is equivalent to the parameter set for the standard access category with $CW_{min} = 31$ and $CW_{max} = 1023$. The priority access category at the AP uses $CW_{min} = 7$ and $CW_{max} = 253$, respectively. All other network parameters are the same as those shown in Table 3.1 in Chapter 3. The results for packet loss and delay are complemented by a voice call quality analysis based on the ITU-T E-model [19]. Recall from Chapter 2, that the E-model uses a so-called R-value to specify the level of call quality, whereby 0 (zero) is the lowest and 100 the highest call quality.

We have devised different scenarios to evaluate the performance of our proposed scheme. An overview of the different scenarios in shown in Table 5.1.

Scenario	Voice codec used in	
	V_d	V_p
1	G.711, 10 ms	G.711, 10 ms
2	G.711, 10 ms	G.729, 10 ms
3	G.711, 10 ms	G.711, 20 ms
4	G.711, 10 ms	G.729, 20 ms
5	G.711, 10 ms	G.723, 30 ms

Table 5.1: VoIP call configuration overview

Throughout this section we will refer to *VoIP call configurations* denoted by $S_i(\alpha N, (1-\alpha)N)$. The configurations indicate the number of voice calls served by each access category at the AP, and the index i indicates the scenario as per Table 5.1. For example, $S_2(5, 2)$ corresponds to scenario 2, where 5 voice calls are served by the V_d (standard) category, and 2 voice calls are served by the V_p (priority) category. Here we consider that all calls served by the V_d category always use a G.711 voice codec with a 10 ms sampling rate, whereas all calls served by the V_p category use a different codec setting, e.g. G.729.

In Chapter 3 we have shown that an IEEE 802.11b infrastructure WLAN can support 6 voice calls using a G.711 voice codec with a 10 ms sampling rate, and that packet loss exceeds the 2% QoS threshold when the 7th call joins the WLAN. Hence the call will enter the *call deterioration phase* as shown in Fig. 5.6, and the user perceived voice call quality is reduced. To mitigate for

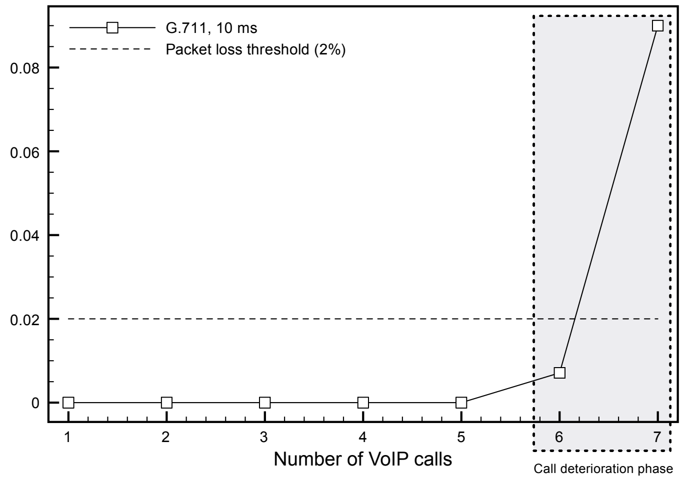

Figure 5.6: Packet loss at the AP obtained analytically for a G.711 voice call including the marked *call deterioration phase*

the loss in quality, here we assume that users use our proposed DCwP scheme and one by one change to a new voice codec until packet loss is below the QoS threshold and an acceptable call quality can be maintained for all voice calls. As a result, the benefits of our scheme are already clear once the WLAN can support 7 voice calls with an acceptable level of call quality. Nevertheless, we also show the total VoIP capacity gain that can be achieved for each scenario.

Scenario 1

In Fig. 5.7 we show the number of calls the WLAN can support for an increasing number of priority calls in scenario 1. It can be seen that in this scenario 5 (simulation) or 6 (analytical model) calls need to be served by the priority queue at the AP, before a performance gain, one additional call, can be achieved. Even though the voice calls served by the priority queue use the same codec as the calls in the standard queue, however, due to the priority setting of the V_p queue, the average service time and the queue utilization is reduced, thus lowering the observed packet loss at the AP, as shown in Fig. 5.8. Our analysis shows that even if the remaining two calls are also served by the priority queue, and further capacity increases cannot be achieved and that the WLAN in this scenario can only support 7 voice calls. Note that the packet loss observed at the AP is the combined loss of both access categories. However, our results show that the priority access category (V_p) does not experience any loss until its queue becomes saturated, thus there is no packet loss in the priority queue. Also note that the simulation matches the analytical results quite well in terms of the number of voice calls the WLAN can support. Similar to our discussion in Chapter 3, in some cases there is a one call difference between the analytical and simulation results. As we have discussed in Section 3.6.2, the slight difference in the number of VoIP calls is due to differences between obtaining the VoIP capacity in the analytical model and the simulation. Recall that packet loss in the analytical model is obtained using Eq. 5.22, whereas $\kappa_{a,sim} = 100 - \cancel{/}\frac{\text{total number of packets received}}{\text{total number of packet sent}} \times \cancel{1}00$ is used in the simulation. Also note that there is a difference in packet loss observed for packet loss above 2% between the analytical and simulation results in some scenarios. We have conducted a variety of experiments, that is simulation and analysis for different parameter settings, to understand the difference. It seems likely that the difference in observed packet loss above 2% may be caused by the difference in obtaining packet loss measurements and the approximation equation itself. Irrespective of the difference in packet loss above 2%, in this thesis we focus on the number of VoIP calls as a measure of performance and as shown the results are in good agreement in terms of the number of calls. However, a 95% confidence interval is shown for results obtained by simulation.

In Fig. 5.9 we show the experienced network delay for calls served by both access categories. Observed that the experienced delay for calls served by the standard queue experience an increasing delay. It is because calls served by the V_d category have to wait for calls served by the V_p queue, and packet loss occurs. However, once a sufficient number of calls is served by the priority queue, the delay drops below the QoS threshold of 60 ms. The drop in queueing delay is because a) the V_d queue only maintains a small number of packets and b) due

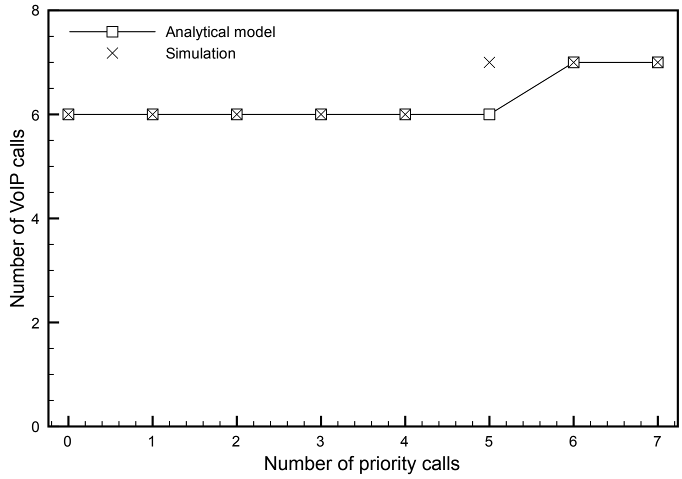

Figure 5.7: Number of VoIP calls (scenario 1)

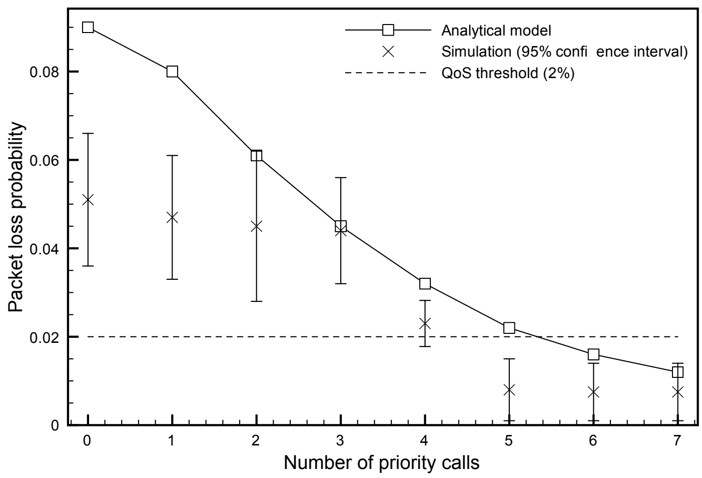

Figure 5.8: Packet loss probability at the AP (scenario 1)

to the different sampling rate of calls served by the V_p queue, there overall level of contention in the WLAN is reduced, and therefore the maximum capacity has not been reached, i.e. the WLAN is not saturated. Also observe, that the experienced delay for all priority calls is well below the QoS threshold.

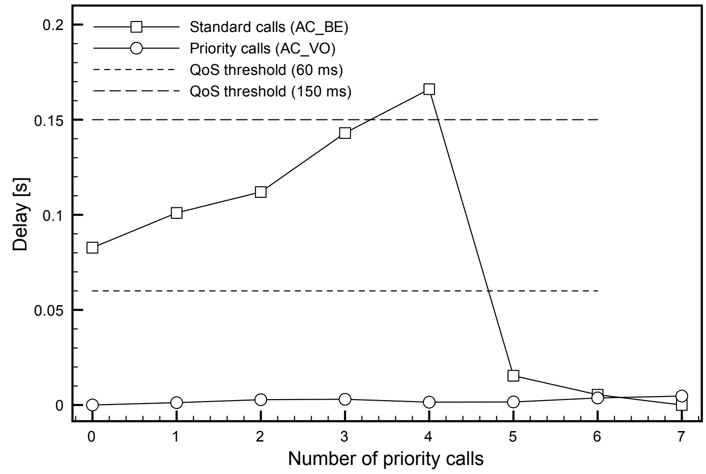

Figure 5.9: Average network delay of standard and priority VoIP calls (scenario 1) [ns-2 simulation]

The reduction in packet loss and the decrease in network delay for VoIP calls served by the V_d access category when there are at least 5 priority calls ($S_1(2,5)$ configuration) is also reflected in the user perceived call quality as measured by the ITU-T E-model. In Fig. 5.10 we show the minimum R-value for the standard and the priority voice calls for an increasing number of priority calls. It can be seen that the priority VoIP calls can always maintain an “excellent” voice call quality ($R_Q = 93$), whereas initially, the standard voice calls experience a “bad” call quality. However, once a sufficient number of users utilized the DCwP scheme with VoIP configurations as per scenario 1, the call quality improves, until all voice calls in the WLAN have an excellent voice call quality. As shown, the call quality improve significantly (“bad” $\rightarrow$ “good”) if at least 5 users adjust their codec ($S_1(2,5)$ configuration). If an additional user adjusts its codec, all calls experience an “excellent” voice call quality.

Additionally, in Fig. 5.11 we show the maximum number of VoIP calls a WLAN can support with and without the DCwP scheme. Here it can be seen that if all VoIP calls are served by the V_p queue at the AP for scenario 1, the WLAN can support a maximum of 7 voice calls which equates to a capacity gain of approximately 16%.

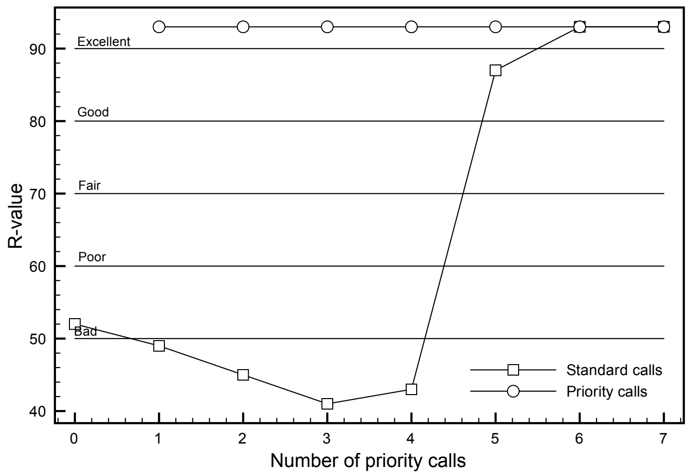

Figure 5.10: Minimum R-value of standard and priority calls (scenario 1)

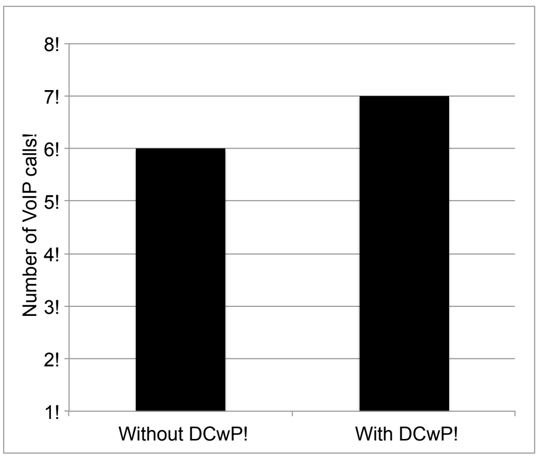

Figure 5.11: Maximum number of VoIP calls with and without the DCwP scheme (scenario 1)

Scenario 2

In Fig. 5.12 we show the packet loss probability at the AP for scenario 2. Similar to the previous scenario, it can be seen that the WLAN can maintain 6 voice calls until a number of users have adjusted to a new codec and thus the voice calls are served by the priority queue at the AP. As shown, the packet loss will drop below the QoS threshold if at least 4 (simulation) or 5 (analytical model) users have adjusted their codec. Note that there is a one call difference between scenario 1 and 2 before the experienced packet loss is below the threshold. This is because even though the adjusted codecs in both scenarios use a 10 ms sampling rate, each call generates 100 packets/s on the downlink, the difference in capacity is due to a smaller payload size of the G.729 voice packets, that is 10 bytes for a G.729 voice packet and 80 bytes for a G.711 voice packet.

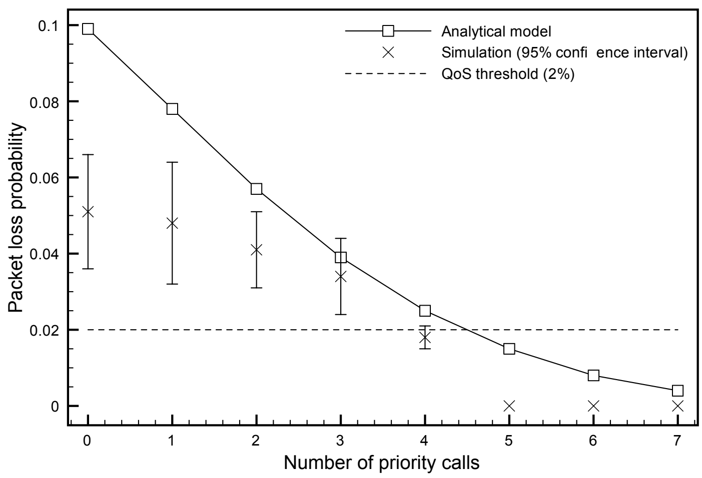

Figure 5.12: Packet loss probability at the AP (scenario 2)

The difference in payload size is also reflected in the total number of VoIP calls the WLAN can support, if all VoIP calls use the G.729 voice codec and are served by the V_p access category. As shown in Fig. 5.13, the overall capacity gain is 2 calls or an increase of approximately 33%.

The experienced delay of the standard and the priority voice calls in scenario 2 is similar to the one experienced in scenario 1 and thus is not shown. The similarity of network delay in scenario 1 and 2 is because the network delay is influenced by the sampling rate rather than the payload size of the individual packet. As we will show in later scenarios, whenever the priority voice calls use a codec with a different sampling rate the experienced delay changes significantly.

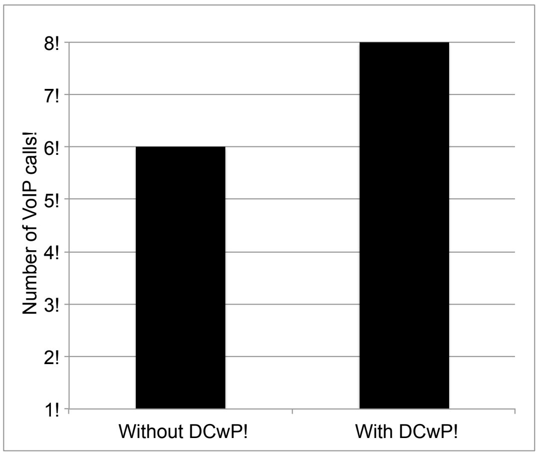

Figure 5.13: Maximum number of VoIP calls with and without the DCwP scheme (scenario 2)

Given that packet loss and delay are comparable in scenario 1 and 2, the user perceived voice call quality also reflects the similarities, as shown in Fig. 5.14. However, note that the maximum voice call quality of a G.729 voice codec is $R_Q = 84$, thus being lower than that of a G.711 voice call. Even though the G.729 voice codec has initially a lower voice quality, in saturated conditions, the call quality is superior. This is because, as shown in Fig. 5.14 the standard voice calls on average have a "bad" quality, whereas the priority calls can always maintain a "good" voice call quality. Only when the packet loss and delay for the standard voice calls is below the threshold, does the quality improves.

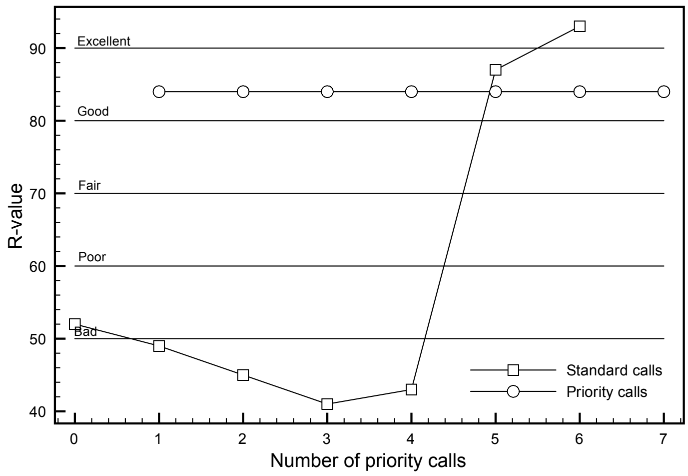

Figure 5.14: Minimum R-value of standard and priority calls (scenario 2)

In the previous two scenarios we considered that users adjust the used codec, without changes to the sampling rate. By using the same sampling rate for all calls, we showed the benefits of our proposed scheme whereby the priority calls experience a significantly reduced network delay and no loss. Hence the priority calls can always maintain a high voice call quality. We also showed that a slight increase in voice capacity can be achieved. In the following, we evaluate the performance of our proposed scheme when the codecs have different sampling rates.

Scenario 3

In Fig. 5.15 we show the packet loss probability at the AP for scenario 3. In this scenario all calls use the G.711 voice codec, however, the voice calls served by the V_d category use a 10 ms sampling rate, whereas voice calls served by the V_p category use a 20 ms sampling rate. It can be seen that the packet loss probability is significantly reduced if a single user adjust their codec. Observe that there is no loss if a second user adjust their codec settings.

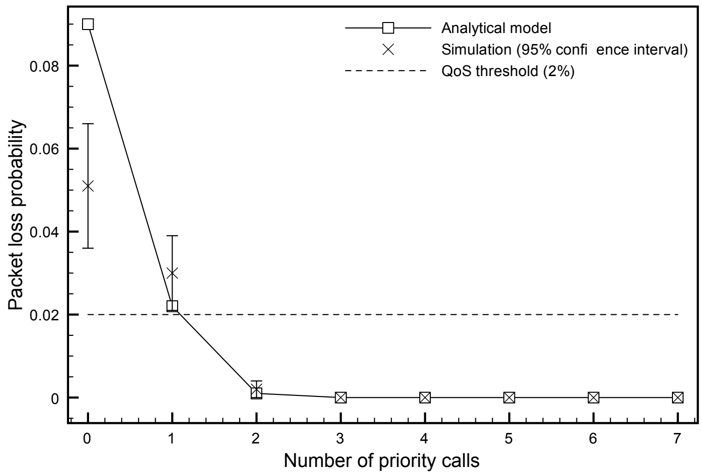

Figure 5.15: Packet loss probability at the AP (scenario 3)

The impact of the different sampling rate is also reflected in the experienced network delay. Figure 5.16 shows the experienced delay of the standard and priority calls. As shown, as soon as two users adjust their codec setting, the experienced delay for the standard voice calls falls below the QoS threshold of 60 ms. This rapid drop in packet loss and delay is due to the different packet arrival rate in each access category. The packet arrival rate per call in the V_p access category (λ^*) is half of the packet arrival rate in the V_d access category (λ). As a result of the lower packet arrival rate, combined with the priority of the V_p queue, the contention in the WLAN is reduced, thus increasing the voice capacity. This increase in capacity is also shown in Fig. 5.17 which shows the maximum number of VoIP calls the WLAN can support in this scenario, when all users adjust the codec as per the DCwP scheme. As shown, the WLAN can then support up to 13 voice calls (a capacity increase of 116%).

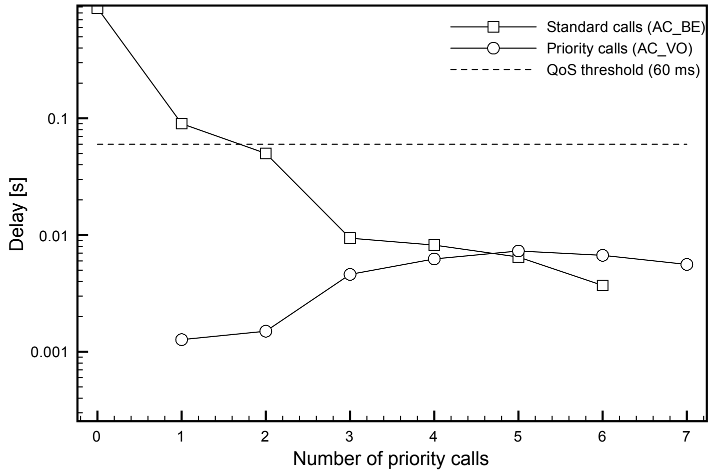

Figure 5.16: Experienced delay of the standard and priority calls (scenario 3) [ns-2 simulation]

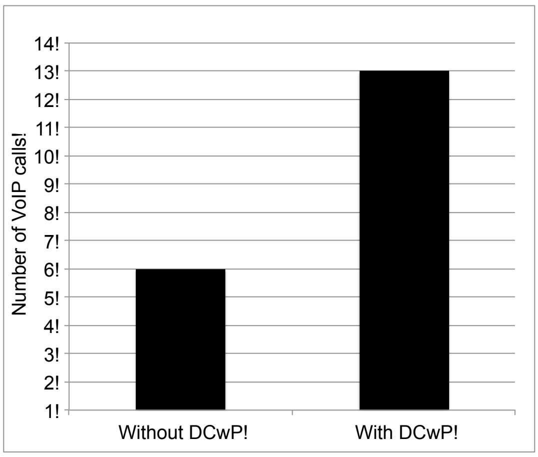

Figure 5.17: Maximum number of VoIP calls with and without the DCwP scheme (scenario 3)

The rapid drop of packet loss and delay is also reflected in the voice call quality, as shown in Fig. 5.18. It can be seen that only for the first two VoIP call configurations the voice call quality of the standard voice calls is considered "bad". However, the voice call quality of the priority call is maximum ($R_Q = 93$), confirming the effectiveness of our proposed scheme. Once two users adjust their codec settings and the packet loss and delay drops below the

QoS thresholds, all users experience a high voice call quality ($R_Q = 93$). Note that the minor difference in delay between the standard and the priority calls will not affect the voice quality.

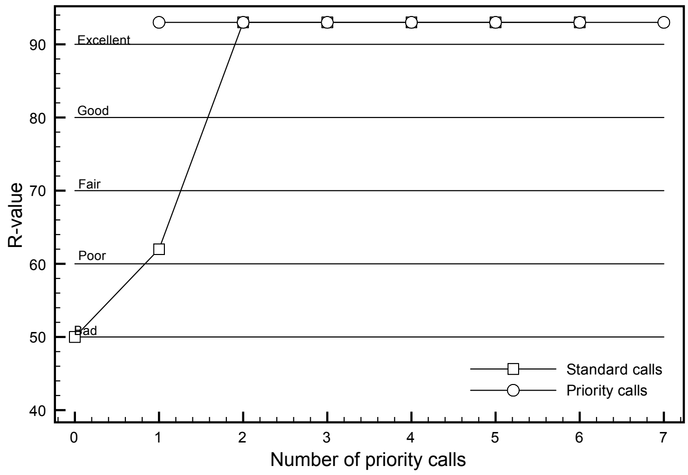

Figure 5.18: Minimum R-value of standard and priority calls (scenario 3)

Scenario 4

In Fig. 5.19 we show the maximum number of voice calls a WLAN can support in scenario 4. As shown, the WLAN can support 6 voice calls using the G.711 voice codec when all users use that codec and all calls are served by the V_d queue. It is also shown that the number of calls is still limited to 6 calls, if a single user switches to a G.729 voice codec. However, as soon as two users use the lower quality G.729 codec, all 7 voice calls can be supported. This increase in capacity is because with an increasing number of priority calls, packet loss at the AP is decreasing as shown in Fig. 5.20. The decrease in packet loss at the AP is due to a lower level of contention in the WLAN caused by a lower packet arrival rate, as previously discussed.

In Fig. 5.21 we show the experienced network delay of the standard and priority voice calls on the downlink. As shown, with an increasing number of priority calls the experienced delay for the standard calls is decreasing. This is because with an increasing number of priority calls, the V_d queue size decreases, thus reducing the delay. Observe that the experienced delay of the priority calls is always well below the QoS threshold. This shows that even though the priority calls use a lower quality codec, the overall call quality will be superior in saturated conditions compared to the standard calls.

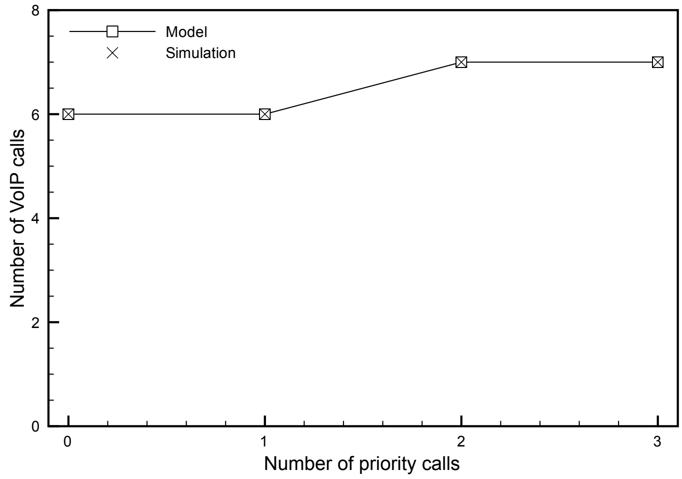

Figure 5.19: Number of VoIP calls if an increasing number of uses change to the G.729 voice codec (scenario 4)

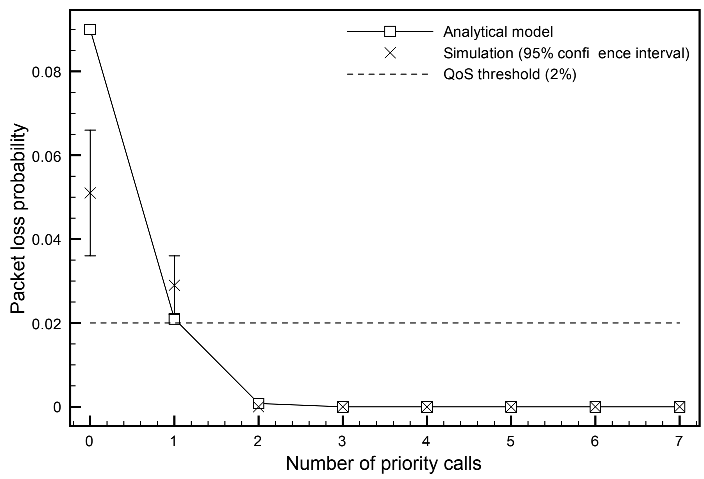

Figure 5.20: Packet loss probability at the AP if an increasing number of users change to the G.729 voice codec (scenario 4)

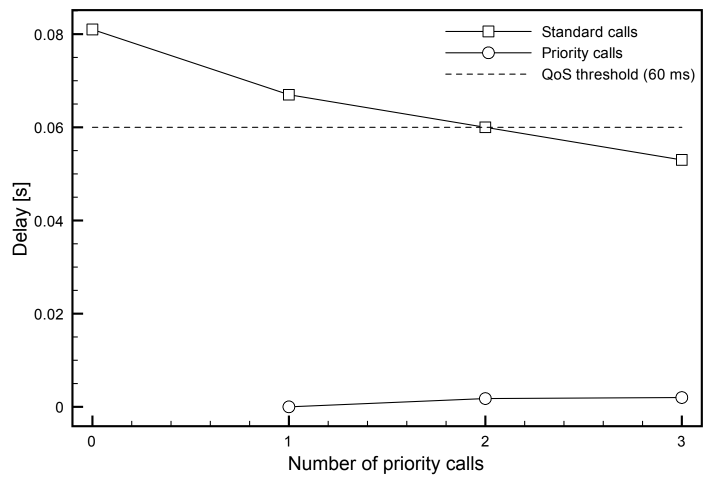

Figure 5.21: Delay of the standard and priority calls if an increasing number of users change to the G.729 voice codec (scenario 4) [ns-2 simulation]

The superior voice call quality of the priority calls is also reflected in the call quality analysis. In Fig. 5.22 we show the $R-values$ for the different VoIP call configurations for this scenario. As shown, the priority voice calls served by the V_p category can always maintain the highest ($R_Q = 84$) voice call quality whenever the network becomes saturated. For example, the standard voice calls can only maintain a voice quality of $R_Q = 83$ and $R_Q = 51$, when there are a total of 6 or 7 voice calls. However, the priority calls constantly maintain the highest $R-value$ as shown in Fig. 5.22 for the G.729 voice codec ($R_Q = 84$) whenever the voice quality of the G.711 calls deteriorates ($R_Q < 93$). Once a sufficient number of users have adjusted the codec, the standard voice calls will maintain a higher voice call quality ($R_Q = 84$) compared to the priority calls. This is because the standard calls do not experience loss or significant delay, thus the G.711 voice codec provides the higher quality.

Similar to the previous scenario, if all calls are served by the priority queue, and the WLAN can then support a much higher VoIP capacity, as shown in Fig. 5.23.

To show the validity of our scheme, we have also conducted measurements in the wireless testbed described in Section 3.5.2 for this scenario. Our scheme was implemented such that the number of calls is increased until packet loss occurs. Once packet loss occurs, the measurement runs for an additional period of time, before the traffic generators gradually reduce the sending rate according to our proposed scheme. In particular, the measurements were conducted as follows:

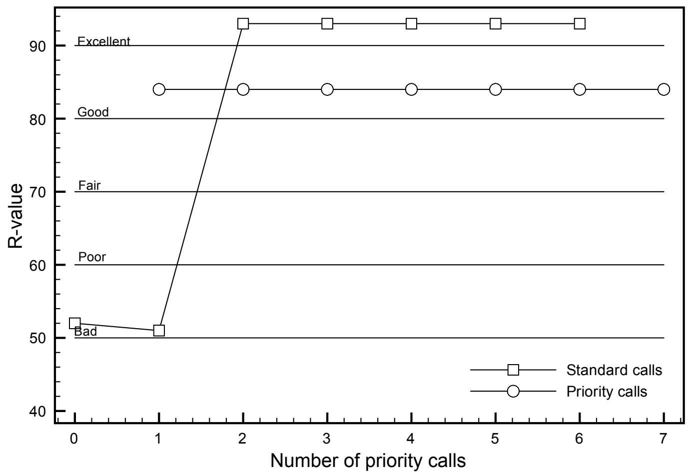

Figure 5.22: Minimum R-value of standard and priority calls (scenario 4)

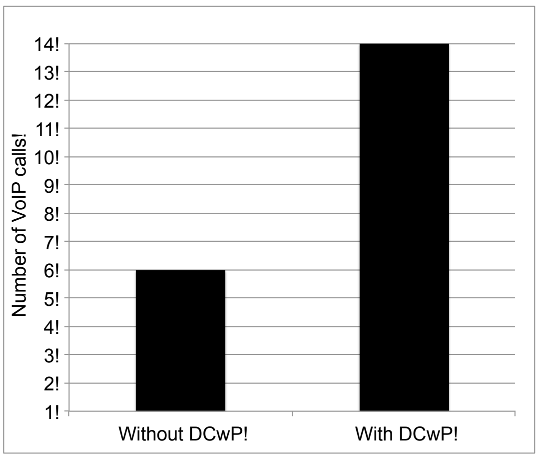

Figure 5.23: Maximum number of VoIP calls with and without the DCwP scheme (scenario 4)

- Start testbed measurement at t_0 [s].
- Add first G.711 VoIP call at t_{10} [s]; add additional G.711 VoIP calls every t_i [s], $i \in \{20, 30, \ldots, 80\}$.
- Maintain VoIP calls during t_{80} [s] and t_{120} [s].
- Adjust first VoIP call to a G.729 voice codec at t_{120} [s]; adjust any subsequent VoIP call at t_i [s], $i \in \{130, 140, \ldots, 180\}$ [s]
- Stop testbed measurement at t_{190} [s]

In Fig. 5.24 we show the average packet loss observed in our WLAN testbed with and without using our proposed DCwP scheme. It can be seen that in this scenario, packet loss occurs when the 6th call joins the WLAN (at 60s) and keeps increasing with any subsequent VoIP call. Observe that whenever the DCwP scheme is used, packet loss decreases when at least two voice calls adjust the codec parameter; the first call at 120s and the second at 130s. The observed packet loss with our proposed scheme after 130 s is below the QoS threshold of 2%. Observe however, that the packet loss is above the 2% packet loss threshold for the remainder of the measurement when the DCwP scheme is not used and all users continue to use the high quality VoIP codec. These results confirm the benefits and effectiveness of our proposed DCwP scheme and are in line with those obtained by our analytical model and simulation.

Scenario 5

In Fig. 5.25 we show the observed packet loss at the AP when several users switch to the low quality G.723 voice codec. The G.723 voice codec supports two different bit rates, 5.3 kbps and 6.3 kbps. In this scenario the 5.3 kbps bit rate was used, hence the codec has a sampling rate of 30 ms with a voice payload size of 20 bytes. Note that the voice call quality for this codec is low. The maximum R-value for this codec is $R_Q = 79$, and as such, the user perceived call quality is only *fair* to *good*. Observe that only a single user needs to switch to this low quality codec, before the packet loss at the AP drops below the threshold of 2%. The improvements are also shown in Fig. 5.26 which shows the minimum R-value of both traffic types. Even though the payload size of a G.723 voice packet is equivalent to a G.729 voice packet (20 ms sampling rate), the results emphasize that the sampling rate of a voice codec is an important factor for the maximum number of voice calls a WLAN can support, as we will show in the next part.

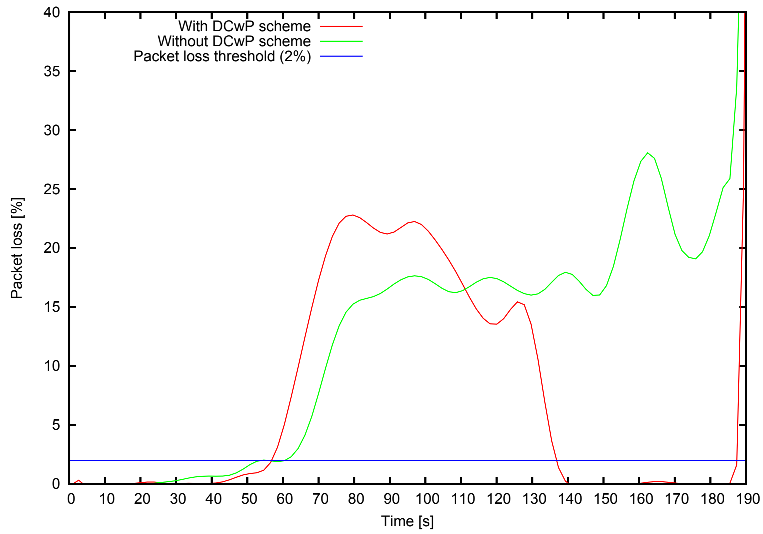

Figure 5.24: Observed VoIP packet loss in the WLAN testbed with and without using the proposed dynamic codec with priority scheme (scenario 4)

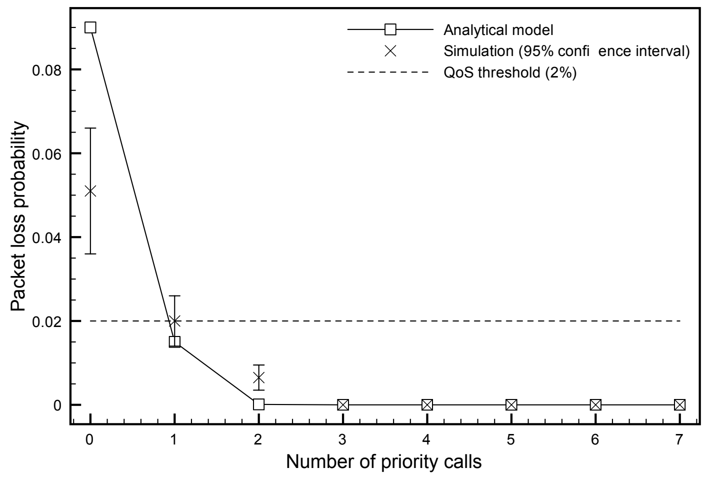

Figure 5.25: Packet loss probability at the AP (scenario 5)

VoIP capacity gain

As we have shown, our proposed scheme allows users who willingly adjust the codec to maintain their call, even if a lower quality voice codec is then used. We have also shown that a voice capacity gain can be achieved with our proposed scheme. For certain configurations, a further capacity gain can be achieved. For example, if all users adjust their codec to the G.723 voice codec, there is only a minimum level of contention in the WLAN. Therefore

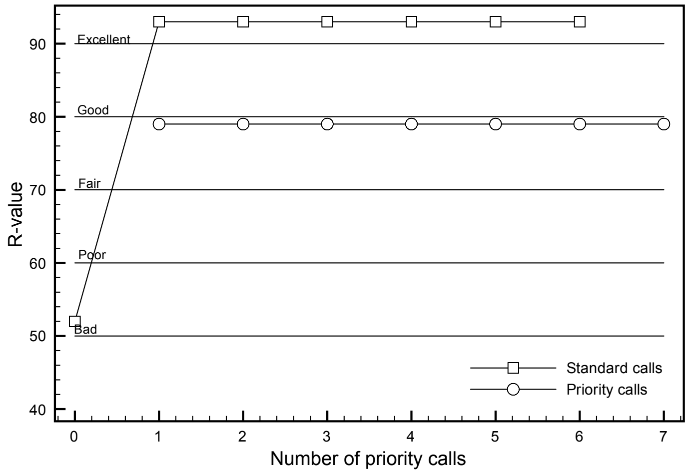

Figure 5.26: Minimum R-value of standard and priority calls (scenario 5)

the WLAN is able to carry additional G.723 voice calls. Using our analytical model, in Fig. 5.27 we compare the minimum and maximum number of VoIP calls that can be maintained for each scenario. Observe that the voice capacity then varies between 6 and 22 calls, thus a performance gain of between 16% and almost 300% can be achieved with our proposed scheme, depending on the scenario. This shows that the proposed scheme can provide a higher capacity gain compared to our previously used solution discussed in Chapter 3. Note that even though the voice capacity can be improved when several users switch to a different codec, the call quality for each individual call is different, depending on the codec used. Also note that we investigate if a further capacity gain can be achieved using different MAC parameter settings later on in this chapter.

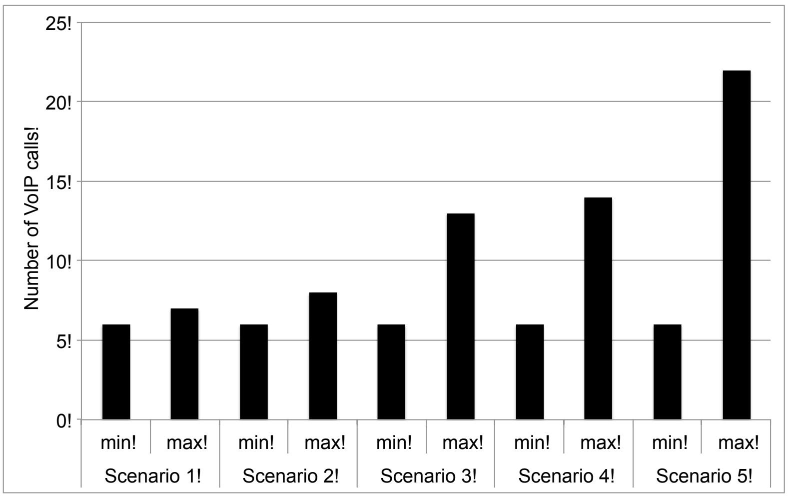

Figure 5.27: Minimum and maximum number of voice calls for each scenario

5.4.1 Single vs. multi-queue AP's

In the previous section we showed the benefits and the performance gain that can be achieved using our proposed scheme. Nevertheless, one might argue that adjusting the voice codec is sufficient, and that the priority queue at the AP will not provide a further performance gain. To show the importance of the multiple access categories at the AP, we compare results of a single and a dual access category system. Here we still assume that whenever the voice call quality is no longer acceptable, a user will adjust the codec as outlined above. However, in the first scenario all calls continue to be served by the V_d category, irrespective of whether the codec has been adjusted or not. In this scenario we consider that a user only adjusts the sampling rate of the codec, rather than switching to a new codec. Thus the priority calls use the G.711 voice codec with a 20 ms sampling rate. It is because we have shown that a sampling rate change has a higher impact than the payload size of a packet. In Fig. 5.28 we show the experienced downlink delay of the standard and priority calls for different voice call configurations when all calls are served by a single access category. Observe that if 1 or 2 users adjust the codec, all voice calls experience a delay above the 60 ms QoS threshold. If at least 3 voice calls adjust the sampling rate, both, standard and priority calls experience a network delay below the threshold.

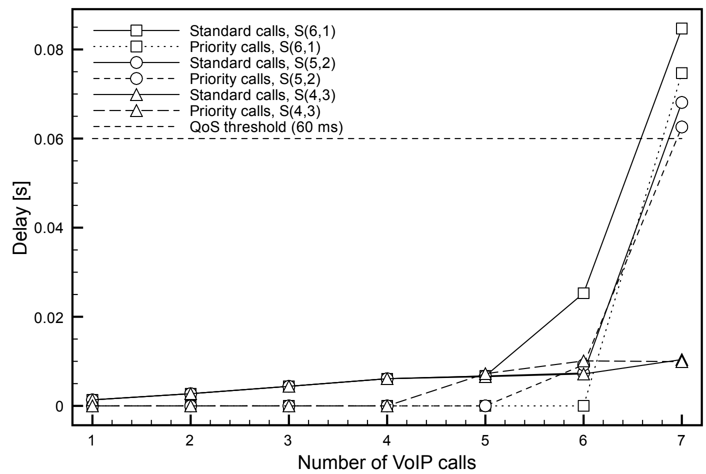

Figure 5.28: Downlink delay of standard and priority voice calls when a single access category is used (ns-2 simulation)

In Fig. 5.29 we show the experienced downlink delay of standard and priority calls for the same VoIP call configurations as above, however, voice calls who adjust the codec will be served by a second, high priority access category at the AP. As shown, for configuration $S(6,1)$ only the standard voice calls experience a delay above the threshold. All priority calls experience a delay below the threshold. This is because the priority calls are served by the V_p access category which allows a more frequent channel access due to a smaller CW_{min} parameter. The results also show that only 2 calls need to adjust their codec before all calls have a network delay below the threshold.

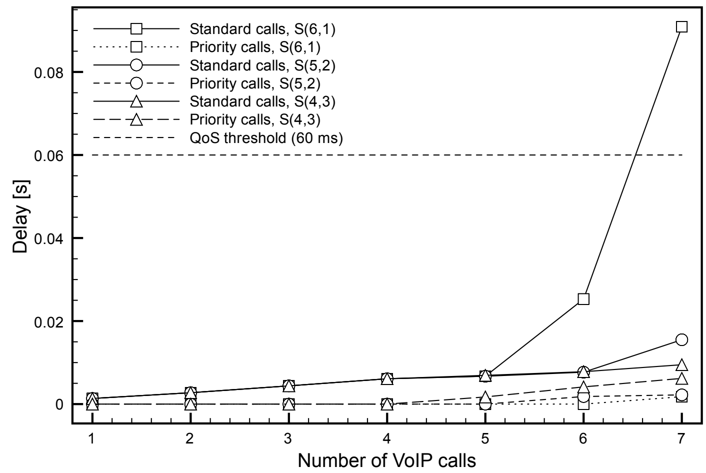

Figure 5.29: Downlink delay of standard and priority voice calls when the V_d and V_p categories are used (ns-2 simulation)

These results show that separate access categories at the AP are beneficial to both types of voice calls by reducing the experienced delay significantly. In particular it can be observed that the experienced delay for users who adjust their codec is well below the QoS threshold at all times, thus allowing a high level of quality.

5.4.2 The impact of internal collisions

In Section 5.2 it was outlined that due to the multiple access categories of the IEEE 802.11 QoS mechanism an internal collision can occur. An internal collision is caused when the backoff process of at least two access categories reaches zero in the same instance. The internal collision handler will resolve the collision by granting channel access to the higher priority access category. The transmission of the lower priority access category is deferred until a further backoff process has been completed. Note that because of this mechanism, a high priority packet can only collide with packets in the channel, whereas a low priority packet can collide with high priority packets internally, and any packet on the channel.

Work which considers the 802.11e protocol and multiple traffic classes, however, often neglects internal collisions. For example, in [120] the authors argue that the internal collision has only a marginal affect and can thus be ignored. Our study however, has a different insight on the impact of the internal collision probability. In this section we will discuss the internal collision probability and show its impact on a variety of different measurements.

In Fig. 5.30 the internal collision probability $\delta\epsilon$for scenarios 1 and 3 is shown. It can be seen that with an increasing number of calls served by the priority queue, the internal collision probability increases. This is expected, because the internal collision probability depends on the queue utilization (see Eq. (5.4)) of the priority queue, which increases with an increasing number of calls served by that queue. As shown, the internal collision probability for scenario 3 is increasing as well, however, it is lower than that of scenario 1. This difference in collision probability is caused by the different sampling rates used in both scenarios. As in scenario 3, a 20 ms sampling rate is used, the packet arrival rate per call in that queue is half of that of scenario 1, thus reducing the priority queue utilization, which subsequently reduced the internal collision probability. Note that $\delta\epsilon = 0$ for both VoIP call configurations $S(7,\epsilon 0)$ and $S(0,\epsilon 7)$. This is because for these configurations all packets are served by a single queue only.

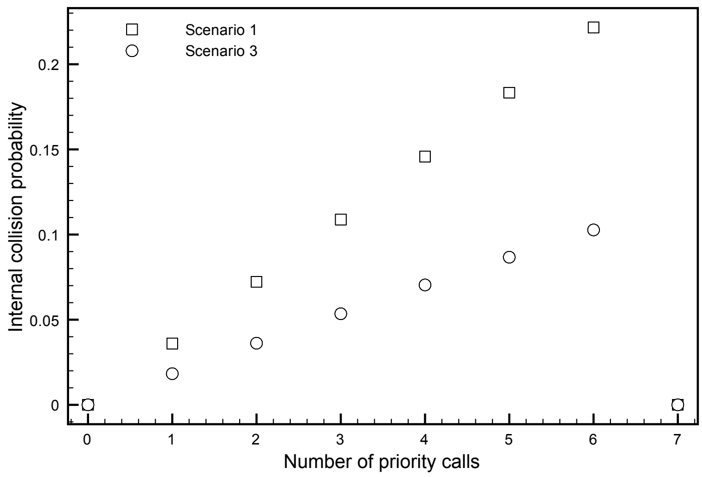

Figure 5.30: Internal collision probability δ for scenario 1 and scenario 3, obtained analytically

A comparison when the internal collision is neglected and considered is shown in Fig. 5.31 for scenario 1. It can be seen that there is a significant difference in collision probability. When the internal collision probability is not considered, in this scenario, the total collision probability is static. This is because the low priority packets can only collide in the medium with packets transmitted by the wireless nodes. As shown, when the internal collision probability is considered, the total collision probability increases linearly with the increase in priority queue utilization.

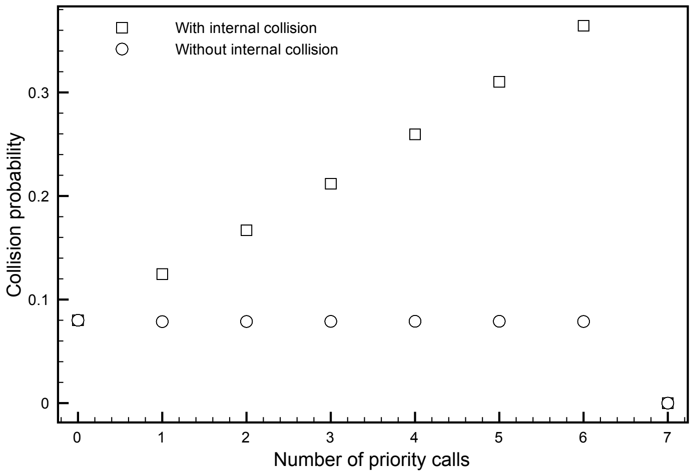

Figure 5.31: Collision probability c_a of an V_d packet with and without an internal collision for scenario 1, obtained analytically

Similar to scenario 1, in Fig. 5.32 we show the total collision probability of the low priority queue for scenario 3, when the internal collision probability is considered and ignored. As in scenario 1, the total collision probability is increasing linearly when the internal collision probability is considered. Note that if the internal collision probability is not considered, it can be seen that with an increasing number of priority calls the total collision probability would be decreasing. This decreasing in collision probability is because the low priority packets can only collide with packets transmitted by the wireless voice nodes, and the total number of packets to be transmitted by the wireless nodes is decreasing, on average, with an increasing number of priority calls. For the latter, this is because the priority calls use a different sampling rate where the packet arrival rate at a station is half of that compared to scenario 1.

The results show that the internal collision probability in general cannot be neglected, as it impacts on the overall system performance. Even though in this scenario the impact of the internal collision is small such that the number of VoIP calls that can be supported does not change, other important metrics for example, the channel access delay or, as shown, the collision probability may change significantly. It is important to note that from the modeling point of

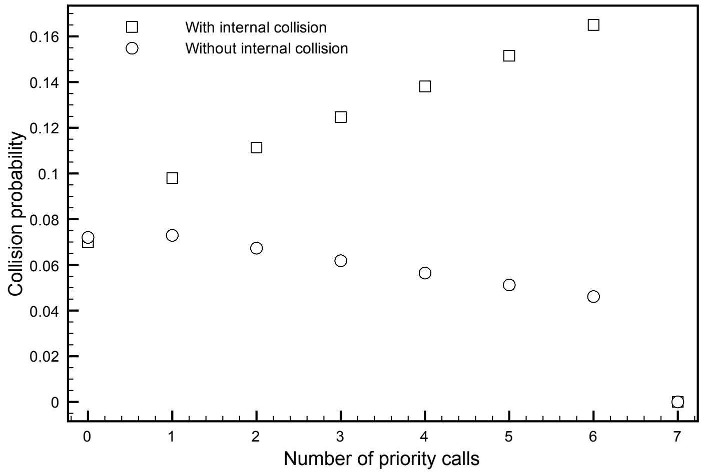

Figure 5.32: Collision probability c_a of an V_d packet with and without an internal collision for scenario 3, obtained analytically

view, a more accurate collision probability is vital because it is an essential part of the fixed-point equations.

5.4.3 The impact of different CW_{min} parameter

In this section we evaluate the impact of different CW_{min} parameter settings for the higher priority access category on the number of calls a WLAN can support. In the previous section, the contention window size for the V_p access category at the AP was set according to the voice access category (AC_VO) of the IEEE 802.11 QoS mechanism, that is $CW_{min} = 7$. In this section we study the impact of different contention window size settings on the performance of our scheme.

In Figs. 5.33 and 5.34 we show the observed packet loss at the AP for different settings of the minimum contention window (CW_{min}) for the high priority access category (V_p) for scenario 1 and 2 obtained analytically. Note that the CW_{max} parameter of the V_p access category was also adjusted in line with the maximum backoff stage m and the retry limit R.

As shown, for $CW_{min} = 3$, a slight performance gain can be achieved. The observed packet loss at the AP drops below the QoS threshold if only 4 users use our proposed scheme, compared to 6 users if $CW_{min} = 7$ for the priority queue at the AP. This shows that fewer users have to switch to a lower quality voice codec, for example to the G.729 voice codec in scenario 2.

In contrast, setting a minimum contention window larger than the default value, for example $CW_{min} = 15$, the observed packet loss at the AP is above the QoS threshold for all VoIP call configurations $S_i(\alpha N, (1-\alpha)N)$. This is because a larger than default contention window increases the random backoff period, and as a consequence increases the average packet service time resulting in an increased queue utilization that ultimately causes packet loss at the AP.

Even though the results shown for both scenarios are the same in terms of the number of users that need to adjust the voice codec, there is a slight difference in packet loss observed in either scenario. Observe that for scenario 2, the packet loss probability is slightly decreasing with an increasing number of priority calls for $CW_{min} = 31$, whereas in scenario 1 the packet loss probability is constant. This is because in scenario 2, the user changes to a G.729 voice codec while maintaining a constant sampling rate of 10 ms. However, as a G.729 voice packet has only 10 bytes compare to the 80 bytes of a G.711 voice packet, the average packet service time reduces, thus reducing the queue utilization, which subsequently reduces the observed packet packet loss slightly. This shows that only setting a smaller than default value of CW_{min} parameter for the V_p queue provides a small increase in performance.

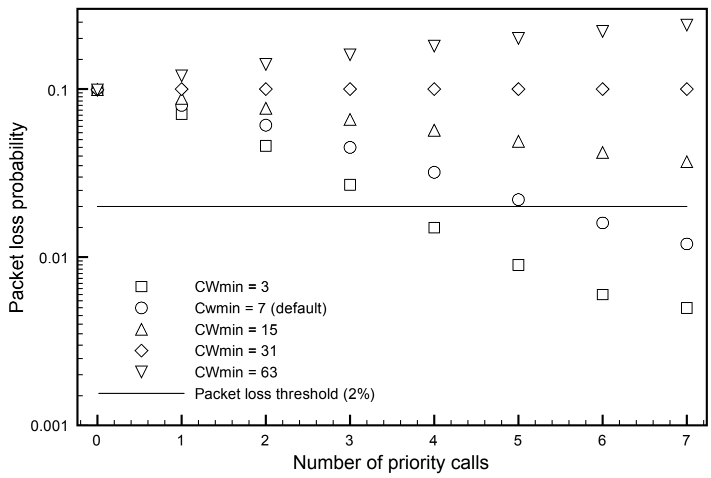

Figure 5.33: Observed (analytically) packet loss at the AP for different values of CW_{min} parameter for the V_p access category (scenario 1)

Setting different contention window size for scenario 3, 4 and 5 does not lead to an increased voice capacity. For example, setting $CW_{min} = 3$ for the V_p access category at the AP will yield equal results to the case when $CW_{min} = 7$. This is because, in these scenarios, the increase in performance due to the different sampling rate of the voice codec is significantly higher than the increase provided by the priority of the voice traffic served by the V_p access category.

The results indicate that the default settings for the contention windows pro-

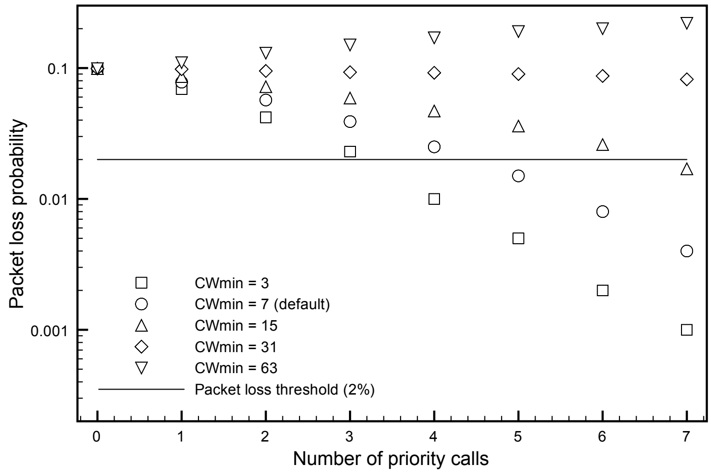

Figure 5.34: Observed (analytically) packet loss at the AP for different values of CW_{min} parameter for the V_p access category (scenario 2)

vide optimal results. Even though a small performance gain can be achieved for a smaller setting of CW_{min} parameter at the V_p queue at the AP, the increase in channel access affects the performance of the wireless VoIP nodes, as shown in Fig. 5.35. Here it can be seen that the collision probability of the wireless voice nodes is higher for $CW_{min} = 3$ than for $CW_{min} = 7$. This is not unexpected, because of the increased channel access, however, as shown, the collision probability is significantly higher (difference of up to 7%) for $CW_{min} = 3$ with an increasing number of priority calls.

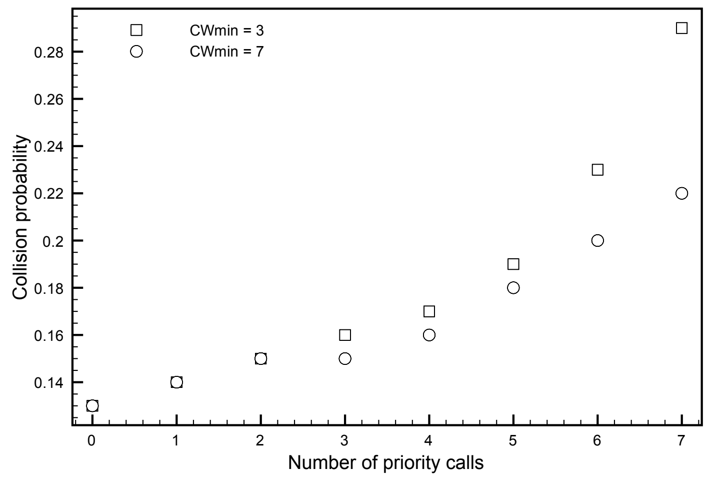

Figure 5.35: Collision probability of the wireless VoIP nodes for different settings of CW_{min} parameter for the priority queue

5.5 Summary

In this chapter, we proposed a novel scheme to exploit a tradeoff between codec quality and priority. In this scheme, users are encouraged to switch to a lower quality codec during periods of high contention and thus enables them to maintain the call. To compensate for the reduction in voice quality, and to give incentive to users to switch to a lower quality codec, a higher priority is given at the AP for those users. As a result of users using a lower quality codec the overall contention is reduced allowing more voice calls to be supported.

We have developed a detailed analytical model to demonstrate the benefits gained from the proposed scheme. Our model offers important insights on the behavior of multi-codec voice streams and the impact of the AP internal collisions. We showed that depending on the codec setting, a capacity gain of between 16% and approximately 300% can be achieved. This significant capacity improvement is without compromising the individual call quality other than using a lower quality codec. By analyzing the observed packet loss and delay, we also showed, using the ITU-T E-model that in most cases the voice calls can maintain a high level of voice call quality. Specifically, we observe that adjusting the sampling rate has a more profound impact on the voice capacity than changing the payload of the voice packet. Furthermore, results obtained from our analytical model matched closely with simulation in terms of the number of voice calls a WLAN can support (i.e. when the loss threshold is not exceeded).

We showed that only using dynamic voice codecs without a prioritized channel access, the voice capacity can not be increased significantly, and that the changes in delay are only marginal.

Furthermore, we showed that using different contention windows can result in an increased performance of the AP, however, we showed that this may affect the performance of the wireless voice nodes. In future work, an optimal design could be sought in terms of other parameter changes.

Also, we showed that internal collisions can affect the overall system performance, and thus cannot be neglected. Even though in the outlined scenario the overall number of calls does not change, from the modeling point of view it is important to consider internal collisions as other network parameters such as channel access delay may be affected.

Overall, our results indicate that our proposed dynamic codec with priority scheme is a viable option to increase the VoIP capacity in an IEEE 802.11 infrastructure WLAN. An advantage of our scheme is that it uses the default parameter settings of the access categories at the AP, and furthermore does not require any changes to hardware, software or the protocol.

6

The impact of TCP flows on the VoIP capacity in IEEE 802.11 WLANs

In the previous chapters we studied the performance of VoIP traffic in IEEE 802.11 infrastructure WLANs. In particular, we were concerned with the limited number of voice calls that can be supported with an acceptable level of quality. To address the limited voice capacity, in Chapter 3 we evaluated a simple solution based on the adjustable $TXOPLimit$ parameter of the IEEE 802.11 medium access control, and showed that a significant ($\approx 100\%$) increase in call capacity can be achieved. Furthermore, in Chapter 5 we proposed a novel scheme to improve the voice capacity without compromising the voice quality, based on dynamic voice codecs and a channel access priority. Our analysis showed that both solutions can provide a significant performance gain in terms of the number of voice calls that can be maintained.

However, our previous analysis considered VoIP traffic only. Nevertheless, in a real-world WLAN the voice flows have to share the channel with concurrent data flows, such as TCP. As a result of the shared channel, the overall level of contention in the WLAN is increased, and therefore we expect that data traffic will impact on the VoIP capacity, such that even fewer calls can be maintained.

In Section 2.3.2 we have outlined some of the relevant literature discussing VoIP and TCP traffic in wireless networks. In [163] for example it is shown that the aggregated voice throughput is slightly reduced when the channel is shared with downlink TCP flows. However, if the channel is shared with uplink TCP flows, the aggregated voice throughput is significantly reduced. Based on these findings, Bellalta et al. [163] proposed a new channel access mechanism using the IEEE 802.11e extension [14] to improve the throughput. Other work, for example by Harsha et al. [104] consider a fixed number of TCP downlink streams and show that the voice capacity is reduced from 12 to 10 voice calls using a G.711 voice codec with a 20 ms sampling rate, whereas Brouzioutis et al. [165] consider a variable number of TCP flows. In particular in [165] it is concluded that the VoIP capacity decreases by two voice calls for each additional data flow. For example, initially the WLAN can maintain twelve voice calls using the same codec as in [104]. If the WLAN is now shard with a single TCP flow, the capacity is reduced to ten calls, and then further decreased to 8, 7 and 5 calls when the channel is shared with 2, 3 and 4 TCP flows. These results clearly highlight that the VoIP capacity is decreased when the channel is shared between VoIP and TCP traffic. Note however, that most of the work studying VoIP and TCP only focusses on a single traffic type, i.e. VoIP. In this chapter we also closely monitor the performance of the TCP traffic.

In this chapter we study the impact of TCP data traffic on the voice capacity in WLAN using ns-2 simulation. This chapter is split into two parts. First we establish a baseline case that is used to compare our proposed solutions to increase VoIP capacity and TCP throughput. In particular, we first confirm some of the results in the literature, in particular that the VoIP capacity depends on a variety of factors, such as the flow direction of the TCP stream and the chosen channel access control, that is DCF or EDCA. Second, we evaluate if the number of VoIP calls and the TCP throughput can be increased if the solutions based on the adjustable $TeXOPLimite$parameter as in Chapter 3 is used. This is followed by an analysis of the performance gain in terms of the number of VoIP calls and TCP throughput, that can be achieved if our proposed dynamic codec with priority (DCwP) scheme is used when the channel is shared between both traffic types.

The remainder of this chapter is organized as follows: In Section 6.1 we provide an overview of our system model. This is followed by the system performance analysis in Section 6.2 that includes the analysis of DCF and EDCA in

Sections 6.2.2 and 6.2.3 and is considered the baseline case. In Section 6.3 we evaluate two different approaches to improve VoIP capacity and TCP throughput based on our solutions presented in Chapter 3 and Chapter 5, before we conclude this chapter in Section 6.4 with a brief summary.

Our contributions in this chapter can be summarized as follows.

1. We investigate the impact of TCP streams on the VoIP capacity in an IEEE 802.11 infrastructure WLAN where the channel access is controlled using the distributed coordination function (DCF) and confirm the reduction in VoIP capacity when the channel is shared with uplink or downlink TCP flows.

2. We investigate the impact of TCP streams on the VoIP capacity if the channel access is controlled using EDCA. We show that the QoS mechanism provided by EDCA provides protection to VoIP and TCP traffic, such that the impact of the TCP flow direction is less restrictive for VoIP and the VoIP streams will not stave the TCP streams. Even though the performance of EDCA compared with DCF is superior in terms of number of VoIP calls and TCP throughput, we show that the default EDCA MAC parameter are not optimal, and that different settings of these parameter can increase both, VoIP capacity and TCP throughput.

3. We investigate whether the solution based on the $TXOPLimit$ parameter used in Chapter 3 can be extended when the channel is shared between VoIP and TCP streams. We show that traffic differentiation using the $TXOPLimit$ parameter for VoIP only provides a tradeoff between VoIP capacity and TCP throughput as the interactions between the two access categories at the AP reduce the overall VoIP capacity. However, we show that using appropriate parameter settings, VoIP capacity and TCP throughput can be increased.

4. Finally, we extend the DCwP scheme proposed in Chapter 5 to the case where the channel is shared between voice and TCP flows. We show that our DCwP scheme is superior in terms of VoIP capacity and TCP throughput compared with the solution based on the traffic differentiation using the $TXOPLimit$ parameter only.

6.1 System Model

Similar to the WLANs considered in Chapter 3 and 5, we consider an IEEE 802.11b infrastructure WLAN consisting of one AP and $N\epsilon$wireless voice nodes. Furthermore there are $M\epsilon = M_u + M_d$ wireless TCP nodes. Each of the $N\epsilon$ wireless voice nodes maintains a full-duplex voice call with a node outside the WLAN. The M_d wireless TCP receivers download an infinitely large file from a source in the wired domain, thus TCP data is carried on the downlink, and TCP-ACK packets traverse the uplink of the WLAN. Similarly, the M_u wireless TCP senders upload an infinitely large file to a destination in the wired network, and thus the network carries TCP data packets on the uplink and TCP-ACK packets on the downlink. We assume basic access using DCF and EDCA is used without RTS/CTS over an ideal channel without interference or hidden terminals and all other parameters are set as shown in Table 3.1 in Chapter 3. Results are obtained using the ns-2 simulator [177] with the EDCA extension of the TU-Berlin [178] with the same parameter settings as in the previous chapters, and the payload size of the TCP data packets is set to 1500 bytes.

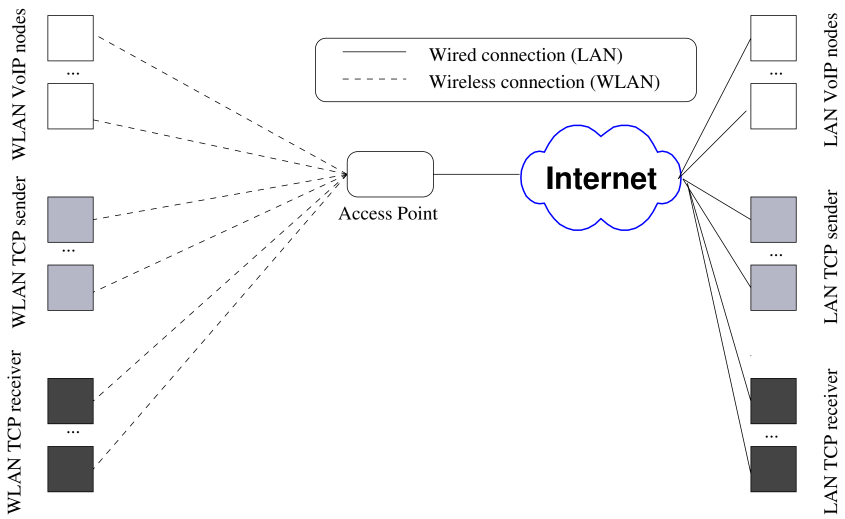

Figure 6.1: IEEE 802.11 infrastructure WLAN topology for VoIP and TCP traffic

6.2 Performance Analysis

6.2.1 Performance analysis: Single vs. multi network metrics

Before we investigate the impact of TCP traffic on the voice capacity, we need a more sophisticated technique for estimating voice capacity of a WLAN than we have used up until now. In the previous chapters we predominantly used packet loss at the AP to determine the voice capacity by setting a loss threshold, beyond which the call quality can no longer be maintained. Using only one network metric works well for a single type of traffic, however, if the network is shared with other traffic, a single indicator cannot be used. In particular, with elastic TCP traffic, loss may not be a suitable indicator, and additional metrics need to be considered. This is because, irrespective of the channel access function, if the AP has to maintain voice and TCP traffic, the delay for each traffic class increases due to the contention between both traffic types. As the TCP traffic rate is governed by the congestion control and avoidance mechanism [40], packet loss at the AP may be low, however, the induced delay increases significantly.

In Fig. 6.2 we compare the voice capacity in an IEEE 802.11 WLAN where channel access is controlled using the distributed coordination function (DCF) and the downlink is shared with a single TCP stream. Observe that the number of G.729 voice calls is 6 when packet loss at the AP is used to determine the voice capacity. However, the voice capacity is reduced to 2, 3 and 4 calls when a delay bound of 60 ms [27], 150 ms [73] or even 200 ms is applied. As in our previous chapters, we apply a packet loss threshold of 2% and require the delay to be less than 60 ms.

6.2.2 Voice vs. TCP: IEEE 802.11 DCF

In this scenario, we consider that the channel access is controlled by the distributed coordination function (DCF). Therefore, both traffic types are served by a single access queue at the AP as "best effort" traffic.

In Fig. 6.3 we show the maximum number of VoIP calls a WLAN can maintain, determined by packet loss, if the voice traffic shares the wireless channel with either downlink or uplink TCP flows. As shown, if the WLAN carries VoIP traffic only, the wireless network can support up to 7 voice calls using the G.729 voice codec with a 10 ms sampling rate. However, if the wireless chan-

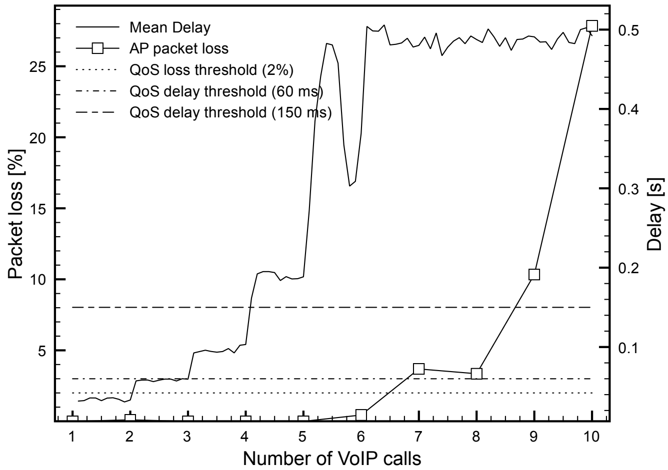

Figure 6.2: Number of G.729 VoIP calls when the VoIP capacity is obtained using packet loss and/or network delay and a single TCP downlink stream (ns-2 simulation)

nel is shared with TCP flows the VoIP capacity is reduced. The reduction in call capacity is caused by the interaction between the VoIP traffic and the TCP congestion control [187–189] and is different depending on the direction of the TCP flows.

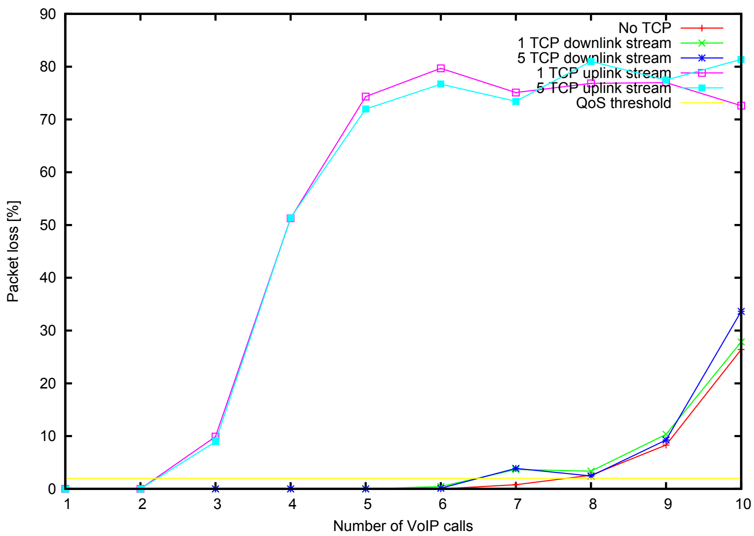

Figure 6.3: Voice call packet loss at the AP for 1 and 5 TCP uplink and downlink flows (ns-2 simulation)

As shown in Fig. 6.3, if the wireless channel is shared between VoIP traffic and downlink TCP streams, the VoIP capacity is reduced to only 6 calls. Recall that in an IEEE 802.11 infrastructure WLAN the AP is a bottleneck limiting the system performance, and in this case the reduction in call capacity is because the single access queue at the AP is shared between packets of both traffic types. As the AP now serves both traffic types, the single access queue at the AP contains packets of both traffic types, and therefore a lower number of VoIP packets can be stored that, as a consequence, reduces the overall number of VoIP calls, as shown in Fig. 6.5. Also note that the VoIP capacity in this scenario is reduced by one call only, irrespective of the number of downlink TCP streams. This is because, if multiple downlink TCP streams share a bottleneck link, each TCP stream approximately obtains an equal share of the available capacity, For example, if $M\epsilon$TCP downlink streams share a total capacity C_{total}, then each TCP stream obtains approximately M/C_{total} of the capacity, which is governed by the TCP flow control of each stream [189]. The equal share of capacity is shown in Fig. 6.4 which shows the aggregated downlink TCP throughput for one and five TCP streams with an increasing number of VoIP calls. Observe that the aggregated TCP throughput is comparable, irrespective of the number of downlink TCP streams. Similar results in terms of the shared capacity between multiple TCP streams has also been shown in [20, 167].

As with an increasing number of VoIP calls the number of TCP packets at the AP is decreasing, the VoIP capacity is reduced by a single call only. The decrease in TCP packets is caused by the TCP congestion control algorithm [187, 188], whereby the TCP sender reduces its sending rate if packet loss is detected. Recall that the TCP sending rate depends on a variety of parameters, for example, the congestion window and the TCP window size. In Fig. 6.6 we show the packet sending rate of a TCP sender with an increasing number of VoIP calls. As shown, initially the TCP sender has a high sending rate, but it is reduced with each new VoIP call, until the TCP stream is almost starved, that is, a packet sending rate of almost 0 (zero) packets. Then the VoIP capacity is limited by the AP as discussed in Chapter 3.

Recall that TCP sender uses the congestion window (*cwnd*) to pace the sending rate. The sender probes the network capacity by gradually increasing its sending rate until packet loss occurs. Once packet loss occurs, the congestion window is decreased and the above process restarts. Therefore the congestion window is used to dynamically adjust the TCP sending rate depending on the available network capacity. In Fig. 6.7 we show the change of the congestion window size with an increasing number of VoIP calls in the WLAN. As shown,

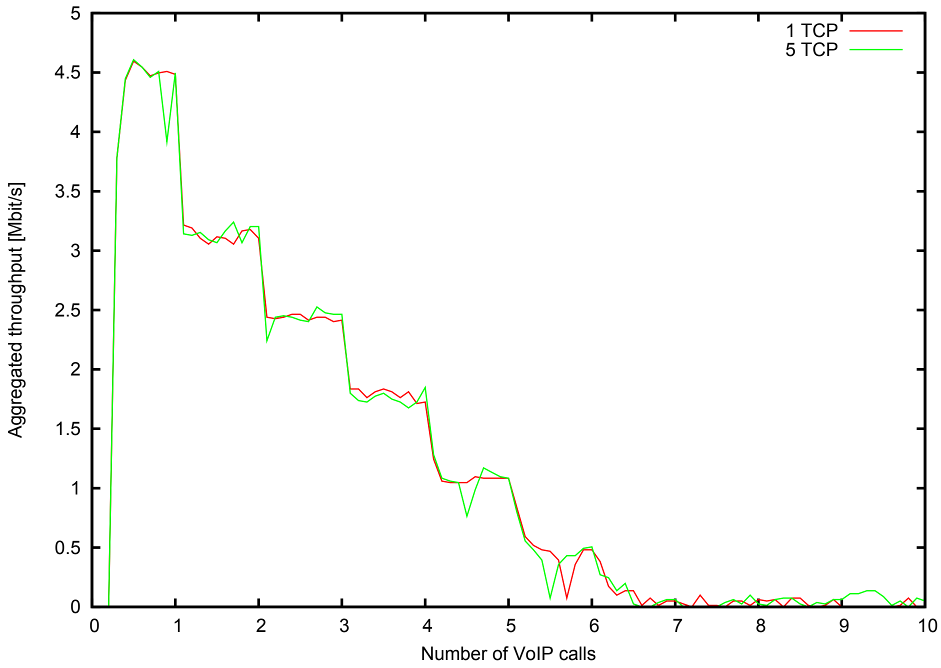

Figure 6.4: Aggregated TCP throughput of downlink TCP streams in an IEEE 802.11 infrastructure WLAN with an increasing number of VoIP calls (ns-2 simulation).

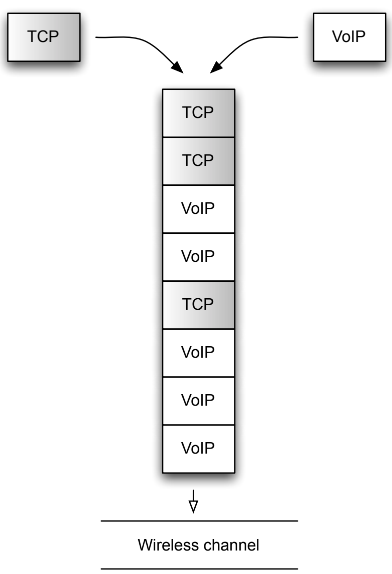

Figure 6.5: Stylized diagram of VoIP and TCP packets sharing a single access queue at the AP

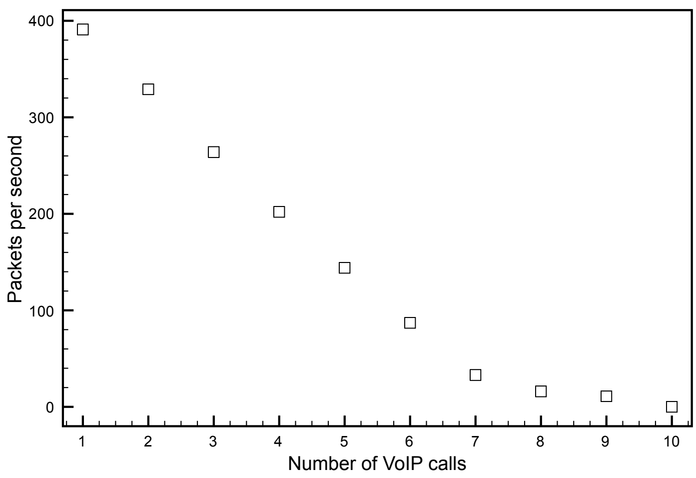

Figure 6.6: Average TCP sending rate (packets/s) for an increasing number of VoIP calls in a DCF controlled IEEE 802.11 infrastructure WLAN (ns-2 simulation)

in the downlink direction, the congestion window is reduced with each additional VoIP call, due to the aforementioned interaction between VoIP traffic and the TCP congestion control. If the channel is shared with uplink TCP streams, the congestion window behaves differently, which we discuss below.

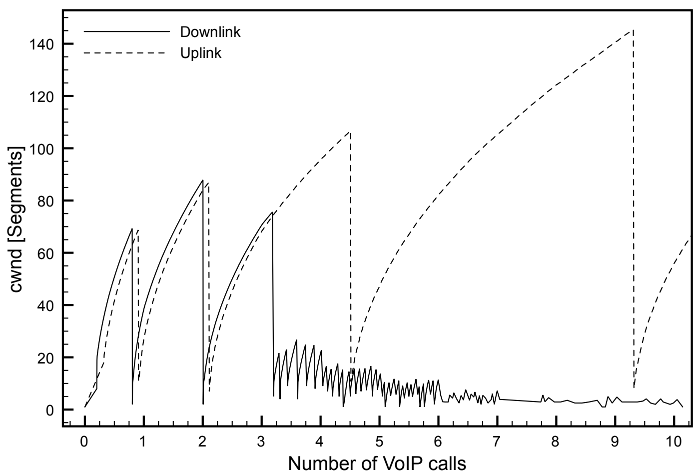

Figure 6.7: Development of the uplink and downlink TCP cwnd with an increasing number of VoIP calls (ns-2 simulation)

As shown in Fig. 6.3, if the channel is shared between VoIP and uplink TCP streams, the VoIP capacity is significantly reduced. This significant reduction in VoIP capacity is due to a limited interaction between the contention of the VoIP packets in the downlink direction and the TCP congestion control restricting the sending rate in the uplink direction. As we have shown in the previous chapters, the AP is a bottleneck in the WLAN in the downlink direction, but as there is not bottleneck in the uplink direction, the TCP senders can transmit the packets at a much higher rate compared to the downlink direction. Additionally, the returning TCP-ACK packets flood the AP access queue, reducing the VoIP capacity to 2 calls only, and can be referred to as AP starvation [163]. Also note that the transmission time of a TCP packets is significantly larger than that of a VoIP packet and the channel is occupied for a longer time duration. Therefore the average packet service time for VoIP packets in the downlink direction increases, which results in an increasing queue utilization that can lead to packet loss.

In summary, we showed that the impact of downlink TCP streams is small compared to the impact of uplink TCP streams. Wheres in the downlink direction the VoIP capacity is reduced by a single call only, in the uplink direction the VoIP capacity is reduced by approximately two-third of its initial capacity.

In Section 6.2.1 of this chapter we argued that multiple metrics should be used when determining the voice capacity. In Fig. 6.8 we show the network delay experienced by the wireless voice nodes for the above scenario. The affect of the TCP flow direction can also be observed for the network delay. As shown, when the wireless channel is used for voice traffic only, the delay rapidly increases whenever the seventh call joins. This shows that at this point the WLAN is saturated. Whenever the WLAN is shared with TCP downlink flows, the experienced delay is significantly higher compared to the the case where no TCP traffic is present. It can be seen that whenever there are downlink TCP flows, the WLAN can only maintain 3, 4 or 5 voice calls using the G.729 voice codec with a 10 ms sampling rate, before the QoS delay threshold of 60 ms, 150 ms or 200 ms is exceeded. Nevertheless, if the channel is shared with uplink TCP flows, the experienced delay increases rapidly when the third voice calls joins the WLAN. In this case, the experienced network delay exceeds the thresholds. This shows that packet loss and delay should be considered to determine the VoIP capacity in an IEEE 802.11 wireless LAN. Note however, that the delay depends on the users tolerance of what is an acceptable delay. In particular, the ITU G.114 standard allows delays of up to 400 ms to be acceptable, and as we have shown in the previous chapters, as the AP is the bottleneck, packet loss rather than delay limits the voice capacity.

The impact of the TCP flow direction, that is the interaction between the voice and TCP flows, is also reflected in the aggregated TCP throughput of the uplink and downlink flows. In Fig. 6.9 it can be observed that in the downlink

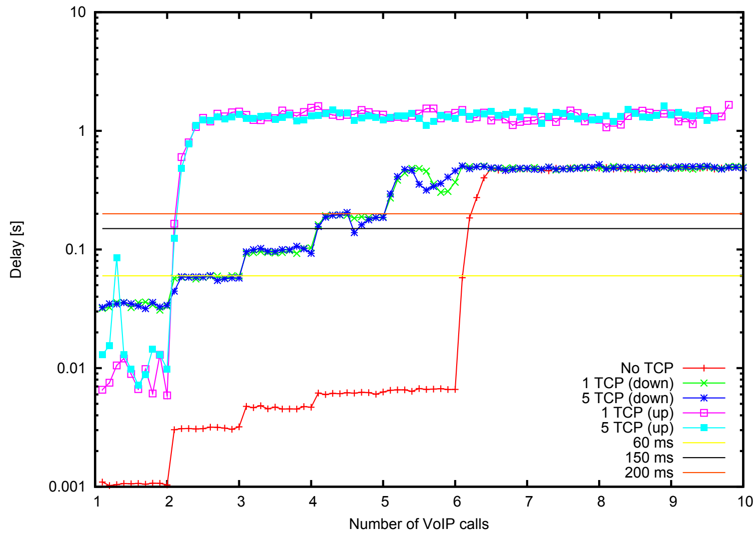

Figure 6.8: Downlink delay experienced by the wireless voice nodes for a different number of TCP uplink and downlink streams (ns-2 simulation)

direction the TCP throughput decreases with an increasing number of voice calls, due to the reduction in sending rate caused by packet loss as outlined previously. In the uplink direction on the other hand, the TCP flows can maintain a high aggregated throughput. The slight decrease in TCP throughput of the uplink flows is caused by the VoIP flows, which increase the level of contention in the WLAN. Nonetheless, as the TCP sender can transmit close to the maximum rate, the TCP throughput is high, whereas the VoIP capacity is low.

6.2.3 Voice vs. TCP: IEEE 802.11 EDCA

Whereas in the previous section we studied the impact of TCP flows on the VoIP capacity in an IEEE 802.11 WLAN when the channel access is controlled using DCF, in this section we study the same scenario, but we assume that the enhanced distributed channel access (EDCA) is used to control access. Recall that EDCA was first introduced in the IEEE 802.11e protocol [14] to allow the provisioning of quality of service to different traffic classes. As discussed in Section 2.2.3, EDCA uses four different access categories to classify traffic. In this scenario we consider that the voice and TCP flows are appropriately classified, and therefore the voice traffic is served by the AC_VO and the TCP traffic is served by the AC_BE access category.

In Fig. 6.10 we show the number of voice calls an IEEE 802.11 WLAN

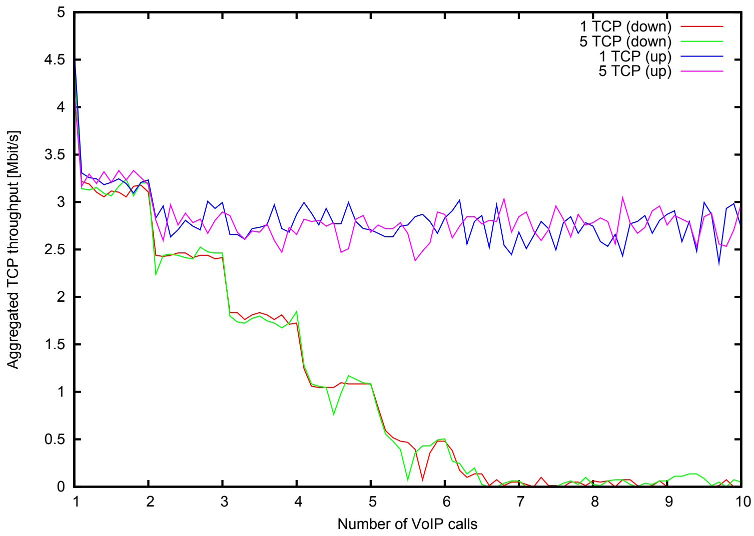

Figure 6.9: Aggregated TCP throughput of TCP uplink and downlink streams with an increasing number of VoIP calls (ns-2 simulation)

can support when the channel access is controlled by EDCA and the channel is shared with uplink or downlink TCP flows. It can be seen that whenever the channel is used to transmit voice traffic only, the WLAN can support up to 9 voice calls using the G.729 voice codec with a 10 ms sampling rate, before the QoS packet loss threshold of 2% is exceeded. Note that the voice capacity is only 7 calls when the DCF is used to control the channel access. The difference in VoIP capacity between DCF and EDCA is because the default EDCA parameter for the AC_VO access category apply a $TXOPLimit$ of 3.264 ms at a voice station in an IEEE 802.11b infrastructure WLAN. Therefore, a station can transmit multiple voice packets when gaining channel access, and as a result, the voice capacity is increased as shown in Chapter 3.

In Fig. 6.10 it can be observed that if the wireless channel is shared between voice and TCP traffic, the voice capacity is slightly reduced, irrespective of the TCP flow direction. In particular, it can be seen that the number of voice calls is reduced by one call only. The reduction in voice capacity is due to the additional channel contention in the WLAN, caused by the TCP traffic. As a consequence the average packet service time for the VoIP packets is increasing, thus increasing the queue utilization in the AC_VO access category and packet loss occurs before packet loss is observed when no TCP flows are present.

The results for the delay are similar to the ones shown in the previous section. As can be seen in Fig. 6.11, whenever there is no TCP traffic, the WLAN

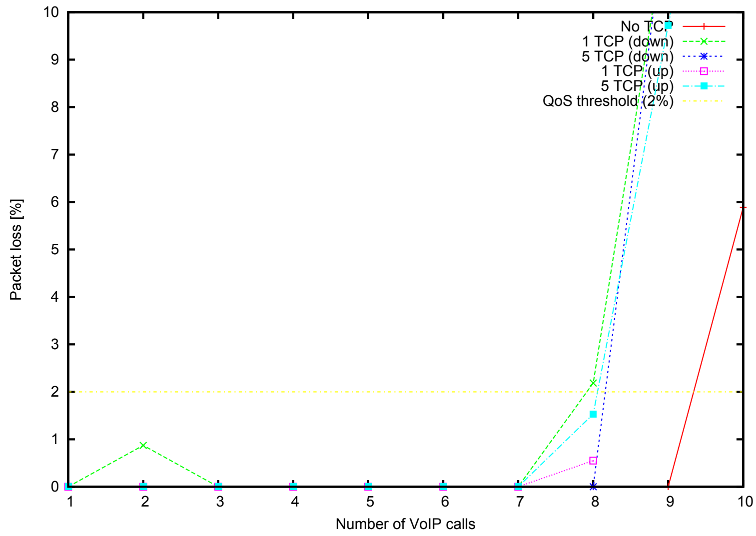

Figure 6.10: Packet loss at the AP using default EDCA parameter for voice and TCP (ns-2 simulation)

can maintain 9 voice calls before the delay experienced in the downlink direction exceeds the delay threshold of 60 ms. In a similar way to the observed packet loss, the delay threshold is exceeded when 7 calls and multiple TCP flows in either direction share the channel.

Applying the default EDCA parameter for voice and TCP traffic also provides protection for the TCP flows. As we have shown in the previous section, if the channel is shared between the VoIP and downlink TCP flows, with an increasing number of VoIP calls, the aggregated TCP throughput decreases, and once the network is saturated the TCP flows are (almost) starved, with a TCP throughput less than 0.1 Mbit/s. However, since in EDCA the AP uses two separate access queues for the different traffic, the TCP flows can gain a higher channel access rate compared to the DCF controlled channel access. As shown in Fig. 6.12, the aggregated TCP throughput that can be maintained is between 0.5 MBit/s and 1MBit/s, irrespective of the TCP flow direction, even if the maximum number of voice calls are carried simultaneously over the WLAN. Note that the lines between the data points is to be understood as a trendline showing the reduction of the average aggregated TCP throughput for uplink and downlink TCP streams.

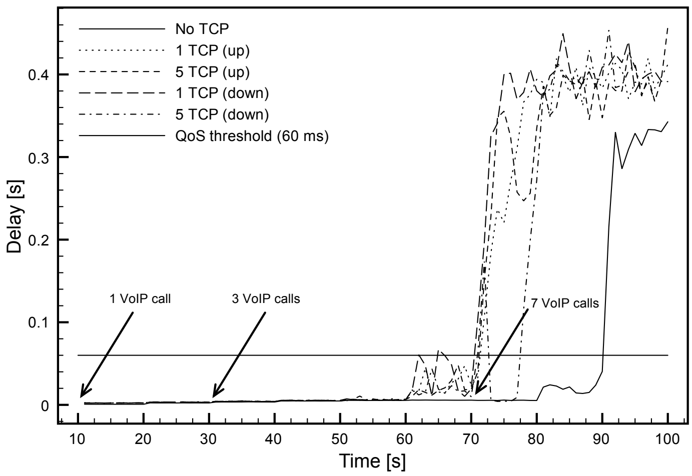

Figure 6.11: Network delay for default EDCA parameter for VoIP and multiple concurrent TCP flows in either direction (ns-2 simulation)

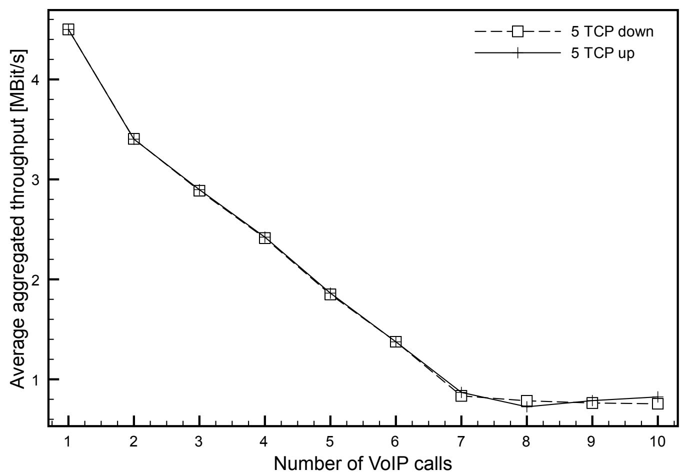

Figure 6.12: Aggregated TCP throughput of uplink and downlink streams using default EDCA MAC parameter settings (ns-2 simulation)

6.3 Improving VoIP and TCP

6.3.1 Voice vs. TCP: The effects of $TXOPLimit$ parameter

In the previous sections we established a baseline case, and showed that using EDCA rather than DCF can improve the voice capacity and also provide protection to TCP data flows. In particular we showed that using the default EDCA parameter that the VoIP capacity is increased by two calls due to a larger $TXOPLimit$ parameter. Also, due to the two different access categories at the AP we showed that the TCP flows can maintain a significantly higher aggregated throughput compared to results showed when DCF is used. However, there have be multiple publications discussing the performance of the default MAC parameter, and suggest means of improving the overall system performance by adjusting the default MAC parameter of a single or multiple access categories [1, 17, 134, 135, 186, 190].

For example, in [134, 135] the authors show that adjusting different MAC parameters can lead to improved system performance by mitigating the delay asymmetry that exist between the wireless nodes and the AP. Also, in [1] the authors show that adjusting the $AIFS$, CW_{min} and $TXOPLimit$ parameters can improve the overall performance of TCP flows in the WLAN. In Chapter 3 we showed that a significant VoIP capacity gain can be achieved if a channel access preference is given to the AP using the adjustable $TXOPLimit$ parameter.

Following our approach of Chapter 3, we assigned a channel access priority to the AP using the $TXOPLimit$ parameter, and we want to investigate its impact on VoIP capacity and TCP throughput. To evaluate the performance, the AC_BE and AC_VO access category at the AP share the same MAC parameter settings, in particular $CW_{min} = 31$ and $AIFS = 50\mu s$, and are differentiated by setting a different $TXOPLimit$ for VoIP only. Therefore, whenever the AC_VO queue wins the channel access, the AP can transmit up to η packets per channel access, as in Chapter 3.

Our results, showed that different access categories with equal MAC parameter settings provide suboptimal performance. In particular for $\eta = 1$ for VoIP traffic, the WLAN can only support 2 voice calls when the channel is shared with TCP flows in either direction. These results are not surprising, and the significant reduction in voice capacity is caused by the interplay of the two queues at the AP. For $\eta = 1$ and equal MAC parameter settings in both access categories, on average, the probability of either sending a voice or a TCP packet is approximately 50%. Thus, for every voice packet sent by the AP, the AP then attempts to transmit a TCP packet in the downlink direction. Note that for simplicity we do not consider internal collisions at the AP. If the channel is shared with uplink TCP flows, the call capacity is also reduced as discussed in Section 6.2.2.

In Fig. 6.13 we show the maximum number of voice calls the WLAN can support when the channel is shared with uplink and downlink TCP flows for selected values of $TXOPLimit$ parameter η. Note that here we only show results if the channel is shared with a single TCP flow in either direction. Considering a single TCP flow is acceptable, because we have shown in the previous section that the number of TCP flows has no affect on the voice capacity, as the TCP flows equally share the channel access [40]. It can be seen that with an increasing value of η, the number of voice calls increases. This result are inline with those in Chapter 3, that is setting a larger $TXOPLimit$ parameter allows the AP to transmit multiple packets per channel access, thus increasing the number of VoIP calls. Observe that the maximum voice capacity is reached for $\eta \epsilon = 5$. Setting a larger value, e.g. $\eta \epsilon = 7$ will not increase the capacity, and setting $\eta \epsilon = 10$ will penalize the wireless voice nodes, as shown previously in Chapter 3. For $\eta \epsilon = 5$ the the network can support 5 and 6 voice calls, depending on the TCP flow direction.

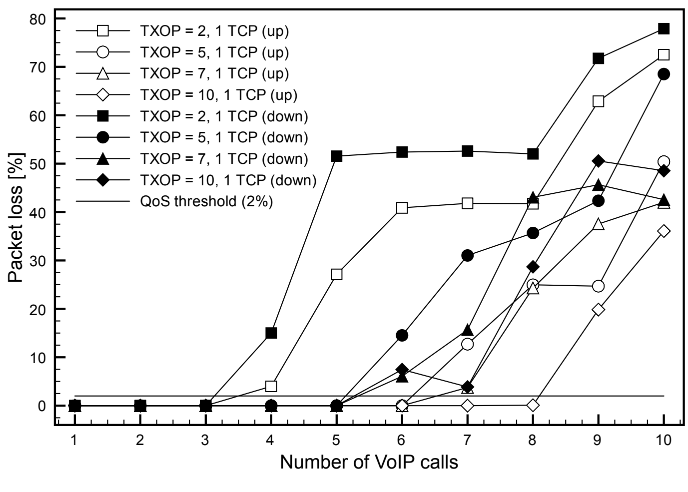

Figure 6.13: Packet loss at the AP for selected values of $TXOPLimit$ parameter η and uplink and downlink TCP flows (ns-2 simulation)

Even though the voice capacity is still lower than compared to using default EDCA parameter, the aggregated TCP throughput on the other hand is improved. In Fig. 6.14 we show the aggregated TCP throughput for different values of the $TXOPLimit$ parameter $\eta \epsilon$ and multiple uplink and downlink

TCP flows. Observe that the lowest TCP throughput that can be maintained is approximately between 1 and 1.5 MBit/s, even though the aggregated TCP throughput decreases for each increased value of $\eta\epsilon$by approximately 250kbit/s. This is significantly higher than the results shown for the default EDCA settings. Therefore, the proposed settings of an equal channel access probability with higher preference for the voice packets of the AP, e.g. $\eta\epsilon = 5$, maximizes the voice capacity in conjunction with aggregated TCP throughput. Therefore, the provided parameter settings are optimal for the scenario when voice and TCP data flows share the wireless channel and provides a tradeoff between VoIP capacity and TCP throughput.

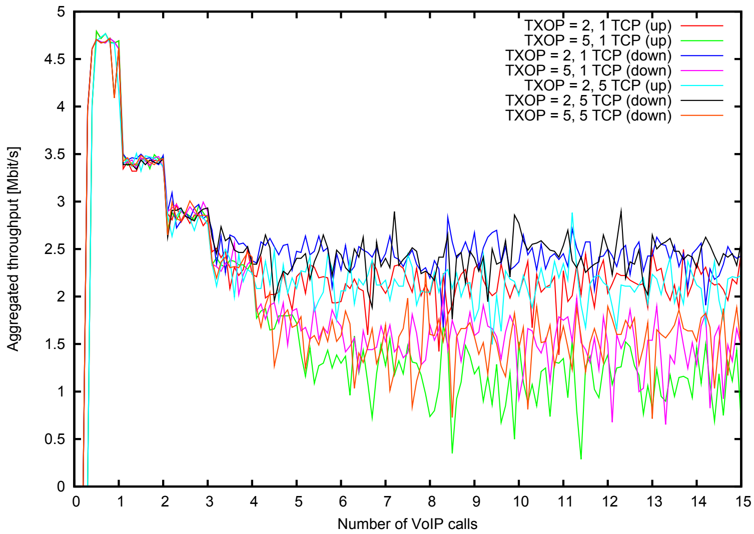

Figure 6.14: Aggregated TCP throughput for selected values of TXOP parameter and multiple uplink and downlink TCP flows (ns-2 simulation)

6.3.2 Voice vs. TCP: The impact of the DCwP scheme on VoIP capacity and TCP throughput

In the previous chapter we proposed a novel scheme to improve the VoIP capacity in WLAN, based on dynamic voice codecs and traffic prioritization. In this dynamic codec with priority (DCwP) scheme we use a tradeoff between call quality and channel access priority to increase the system performance. As we have shown, the DCwP scheme can provide a voice capacity gain of almost 300% for some scenarios, with a high level of voice call quality.

In this section we investigate if a similar performance gain can be achieved if the channel is shared with concurrent TCP flows. Beside the increase in VoIP capacity, we also want to study if an increase in TCP performance, i.e. TCP throughput, can be achieved.

Recall that the DCwP scheme is implemented by way of using two different queues at the AP, a standard queue and a priority queue, denoted by V_d and $_p$, respectively. All high quality voice calls use the G.711 voice codec with a 10 ms sampling rate, and initially all calls as well as the TCP flows are served by the V_d queue at the AP. As with an increasing number of VoIP calls at some stage the user perceived call quality can no longer be maintained as the network becomes saturated, an individual user can then chose to adjust the voice codec to a lower quality codec, for example, adjust from the G.711 voice codec with a 10 ms sampling rate to a G.729 voice codec with a 20 ms sampling rate. As per the scheme, to provide an incentive to the user to adjust to a lower quality voice codec, the voice packets of such calls are then served by the priority queue at the AP, thus guaranteeing throughput and high call quality. Note that in this section we will focus on the two scenarios 4 and 5 defined in Chapter 5, because those scenarios yield the highest performance gain in terms of numbers of voice calls.

We have shown previously that an IEEE 802.11b infrastructure WLAN can only maintain 6 voice calls using the G.711 voice codec with a 10 ms sampling rate. Similar to the result shown earlier in this chapter, once the wireless channel is shared with either TCP uplink or TCP downlink traffic, the voice capacity is reduced, as shown in Fig. 6.15. In particular, if the channel is shared between voice and downlink TCP flows, the VoIP capacity is reduced by one call, and the WLAN can now only 5 calls. If the channel is shared with uplink TCP flows, the capacity is reduced to two voice calls only, due to reasons discussed earlier on. Therefore the benefits of our scheme can be observed if the WLAN can support a minimum of 6 voice calls if the channel is shared with downlink TCP flows, or more than 2 voice calls if the channel is shared with uplink TCP flows.

In Fig. 6.16 we show the number of VoIP calls that can be supported in scenario 4 with an increasing number of priority calls, when the channel is shared with downlink TCP flows. It can be seen that the results are equivalent to those shown in Chapter 5, and that at least two users need to adjust the voice codec to lower the level of contention in the WLAN, and thus increase the number of calls that can be maintained. As shown, if the WLAN is shared with TCP downlink flows, the network can support up to 6 voice calls if at least 2

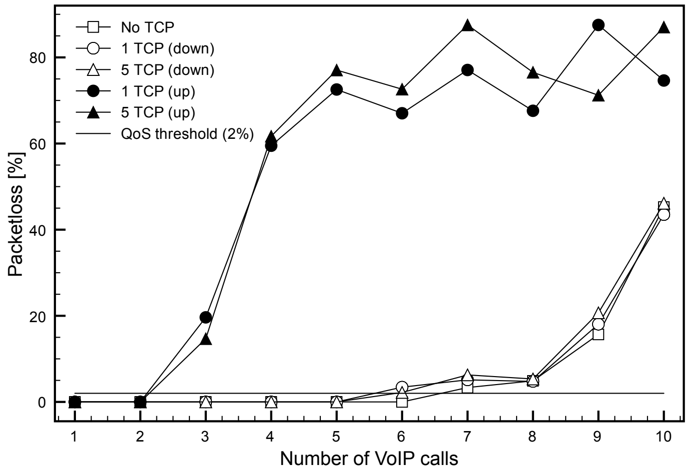

Figure 6.15: Number of G.711 voice calls when the wireless channel is shared with TCP downlink flows without the DCwP scheme (ns-2 simulation)

voice calls are served by the V_p queue at the AP. This result shows that the DCwP scheme also provide a capacity gain if the channel is shared with TCP downlink flows.

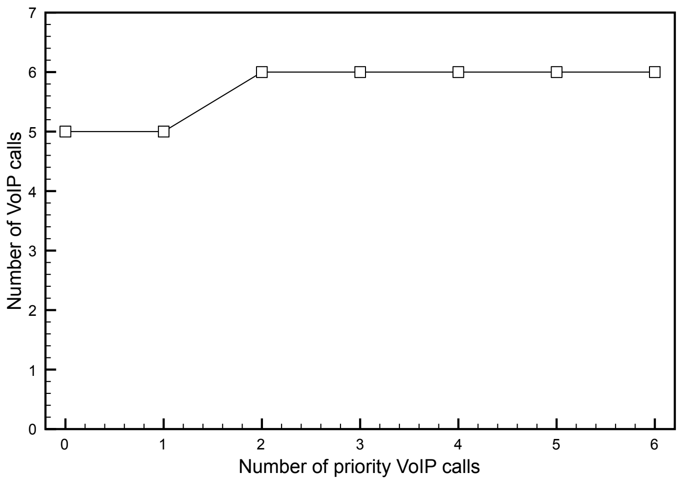

Figure 6.16: Number of VoIP calls for an increasing number of priority calls using Scenario 4 as in Chapter 5 (ns-2 simulation)

Even though the DCwP scheme works well if the channel is shared with downlink TCP flows, as the number of high quality calls can be increased, the scheme does not provide a further call capacity improvement if the wireless channel is shared with uplink TCP flows.

Using our DCwP scheme, we also want to investigate if the proposed scheme provides some benefits to the TCP flows, i.e. an increased TCP throughput.

In Fig. 6.17 we show the downlink TCP throughput for the scenario described above. Note that initially, there is no VoIP traffic, and that each individual call is added to the WLAN every 10 s, for example, the first call is added at 10 s, the second at 20 s and so forth, until the 6th call joins at 60 s. It can be seen that with an increasing number of VoIP calls, the TCP throughput is decreasing. This is in line with results presented earlier in this chapter. However, it can be observed that for the different configurations (denoted as $S_i(\alpha N, (1-\alpha)N)$ as in Chapter 5), the minimum TCP throughput is increasing. For example, for the configuration $S_4(6,0)$ (6 standard calls, 0 priority calls) the TCP throughput is starved as the network is saturated. However, if the WLAN carries for example three standard and three priority voice calls ($S_4(3,3)$) (3 standard calls, 3 priority calls), the minimum throughput that can be maintained is approximately 750 kbit/s. Further increasing the number of priority calls in the WLAN will further increase the minimum TCP throughput up to approximately 2.5MBit/s for the $S_4(0,6)$ configuration. This increase in TCP throughput is because of the reduced number of voice packets in the V_d queue, thus reducing the TCP packet loss, and the TCP sender can maintain a higher sending rate. Comparing the TCP throughput results with those presented in the previous section where we used the adjustable $TXOPLimit$ parameter, it can be seen that a slightly higher and more consistent TCP throughput can be achieved. This shows that the DCwP scheme in terms of TCP throughput is superior to just using the $TXOPLimit$ solution and that for certain scenarios an even higher VoIP capacity can be achieved if the channel is shared between VoIP and downlink TCP traffic.

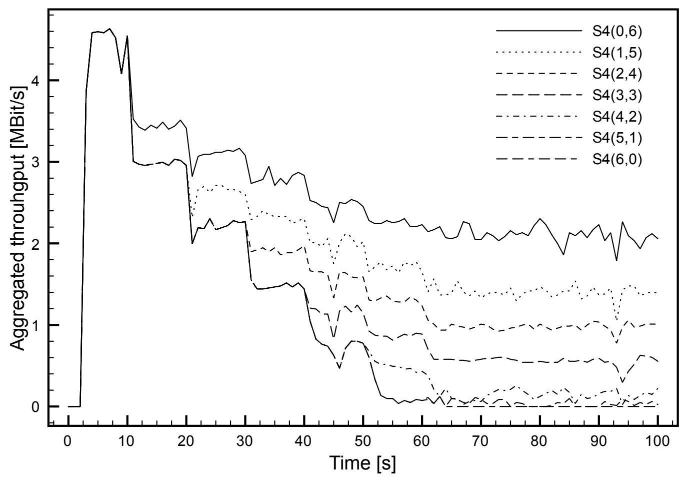

Figure 6.17: TCP throughput for scenario 4 when the channel is shared with TCP flows (ns-2 simulation)

The results for scenario 5, whereby a user switches to the G.723 voice codec, are similar the results above. However, a slightly higher TCP throughput can be achieved. This is because the G.723 voice codec has a 30 ms sampling rate, as as such generates a lower number of packets, thus allowing an increased number of TCP packets in the WLAN, and as a result increases the TCP throughput.

6.4 Summary

In this chapter we studied the impact TCP uplink and downlink flows have on the number of voice calls that can be maintained in an IEEE 802.11 infrastructure WLAN. We confirmed that the voice capacity is reduced by one call when the channel is shared with downlink TCP flows, but whenever the channel is shared with uplink TCP flows the number of calls that can be maintained is significantly reduced. This difference in call capacity is caused by the interaction and independence of the TCP flows with the voice flows at the AP and the wireless nodes. By analyzing the voice capacity in an a DCF and EDCA controlled WLAN, we showed that an increased capacity can be achieved when EDCA was used. This was not unexpected, as EDCA has been designed to provide quality of service to different traffic categories in wireless networks.

Following our approach in Chapter 3, we investigated giving preference to the downlink voice streams using the adjustable $TXOPLimit_e$ parameter at the AP. By applying equal MAC parameter at both access queues at the AP, and only varying the $TXOPLimit_e$ parameter η_e for the voice traffic, we showed that the interaction between the two access queues at the AP severely limit the number of voice calls. However, then applying an increasing value of η_e for the voice traffic only showed that the voice capacity gain can be achieved. Even though the total capacity was one call short of the voice capacity that could be achieved with the default EDCA parameter, the overall TCP throughput was increased. Therefore setting a large value of η_e at the AP for the voice traffic is a tradeoff between voice capacity an TCP throughput.

We showed that our proposed scheme in Chapter 5 can also provide great benefits to the user if the channel is shared with TCP flows. We showed that the voice capacity can be increased if a user adjusts to a lower quality voice codec but the packets are served by a higher priority access category. In particular we showed that the WLAN can support a minimum of 6 voice calls in conjunction with TCP downlink flow if a least two users adjust the codec. The maximum voice capacity increase here is in line with the results reported in Chapter 5. However, considering also the TCP throughput, we see that our proposed solution is superior to only using the $TXOPLimit$ limit parameter for the voice traffic. Whereas in the solution utilizing η for voice only, the TCP throughput decreases with each increase in η, using the DCwP scheme we see that the minimum TCP throughput increases with each user adjusting to a different voice codec. These results show that the DCwP scheme can improve the voice capacity without compromising the call quality of all voice calls, and furthermore improves the minimum TCP throughput.

Perhaps the key lesson from this chapter is that optimizing throughput of both VoIP and TCP is challenging. There are complex interactions between the parameters used to configure the IEEE 802.11 MAC and TCP. The work in this chapter has explored some of these interactions.

7
Conclusion

The aim of this thesis was to provide an in-depth analysis of VoIP in IEEE 802.11 infrastructure WLANs to gain a better understanding of the subpar performance of VoIP in wireless environments. We approached the performance problem by developing a detailed analytical model that captured the complex interactions between the VoIP traffic and the contention based channel access using EDCA of the IEEE 802.11 Medium Access Control protocol. With this model we first investigated the VoIP performance in terms of the number of voice calls with an acceptable level of call quality, and then presented different solutions to improve the performance inline with our requirements for an ideal solution as outlined in Chapter 1.

Considering an IEEE 802.11 infrastructure WLAN with EDCA basic access, we showed that a significant increase in VoIP capacity was achieved when an access priority was assigned to the AP to alleviate the AP bottleneck problem using the QoS mechanism of the IEEE 802.11 protocol. In our analysis we identified optimal MAC parameter settings that increased the number of VoIP calls without compromising the overall system performance, for example, excessive delays at wireless VoIP nodes. In particular, we showed that some suggested parameters in the relevant literature are too aggressive causing the overall performance of the system to be reduced. In particular we highlighted that too aggressive parameter settings can severely affect the performance of the wireless voice nodes and that the bottleneck in the WLAN can shift from the AP to the wireless voice nodes.

Our analytical model also allowed us to study the buffer requirements for VoIP in WLANs. Our analysis based on traffic prioritization at the AP significantly increased the number of VoIP calls that can be maintained. However, we also showed using our model and simulation that if the number of calls kept increasing, at some stage, packet loss at the AP occurs, because the packet arrival rate at the AP exceeded the capacity of the AP in terms of service rate and buffer space. Our analysis of the buffer requirements revealed that the VoIP capacity in an IEEE 802.11 infrastructure WLAN is independent of the buffer size and that there exists a minimum buffer size that maximizes the VoIP capacity. To substantiate our claim of the buffer size independence, we modified our analytical model such that an infinite buffer space was considered, and we showed that the results obtained with this model are a close match to our results obtained when a finite buffer size is considered. As part of our modified analytical model, we presented a closed-form expression for the VoIP capacity in a WLAN as a simple means to obtain the number of calls that can be supported, rather than deriving the capacity based on a fixed-point formulation or a complex Markov-chain based model as used in some of the related literature.

Based on our finding of the buffer size independence and the aforementioned closed-form expression we proposed a novel way of obtaining the number of VoIP calls that an IEEE 802.11 infrastructure WLAN can support, and proposed the *VoIP capacity approximation*. This novel approximation equation allows us to provide a further insight into the interaction between VoIP traffic and the contention based channel access in an EDCA controlled WLAN. In particular, this approximation allows us to analytically confirm the asymptotic VoIP capacity that had been identified in our initial discussion about the VoIP capacity in conjunction with the MAC parameter optimization.

Besides using traffic differentiation to increase the number of VoIP calls that can be supported in a common IEEE 802.11 infrastructure WLAN, we proposed a novel scheme based on dynamic voice codecs and traffic prioritization at the AP to exploit a tradeoff between call quality and traffic priority. We showed that the proposed *dynamic codec with priority* (DCwP) scheme provides a superior VoIP capacity without compromising the individual call quality compared with the previously used solution based on traffic differentiation only. A particular focus of our analytical model was to accurately capture the internal collisions between the different access categories at the AP, and we showed that internal collisions cannot be neglected as is done in some of the related literature. Additionally, we implemented the ITU E-model to demonstrate the higher call quality our proposed scheme can provide.

Furthermore, we extended our analysis of VoIP in IEEE 802.11 WLANs to show the impact of TCP traffic on VoIP capacity. Even though the impact of TCP traffic on VoIP in WLANs had been investigated in some of the relevant literature, in this work we presented a further analysis when VoIP and TCP traffic share the channel in DCF and EDCA controlled WLANs. Our analysis was focused on the improvement of systems performance, and we showed that our proposed DCwP scheme is superior in terms of increased VoIP capacity and TCP throughput if compared with results based on traffic differentiation only.

7.1 Future work & Outlook

This work has explored the complex interaction between VoIP traffic, the Medium Access Control of the IEEE 802.11 protocol and how they affect the performance of VoIP throughput. We have developed a good understanding of these interactions and how VoIP capacity in WLAN can be significantly increased. However in the course of the research work, additional questions have been identified. Here we present a brief overview of some of the potential research questions.

IEEE 802.11n: How well can the new high speed IEEE 802.11 protocol support VoIP traffic? An initial analysis is presented in [107], however, a full investigation into the different frame and ACK aggregation techniques that yield the highest VoIP and overall system performance is yet to be undertaken.

Performance of dynamic codecs: Our analysis of dynamic codecs was based on the G.729 and G.711 voice codecs with known properties and traffic patterns. However, how well does a dynamic voice codec, such as SILK in Skype, work in WLANs, and will the codec perform as well as predicted in our proposed DCwP scheme?

VoIP vs. streaming video: Throughout this thesis we have demonstrated that our analytical approach is versatile and that our analytical model can be used to investigate different scenarios and traffic types, for example streaming video traffic. Streaming video is an important emerging service, due to its potential to replace current television services and incorporate streaming video with other consumer relevant services, for example, social TV. However, the open questions here are for example, what impact has streaming video traffic on VoIP in WLANs? Are the default QoS parameter settings sufficient to provide an optimal system performance and cope with high definition TV streams?

Implications of internal collisions for different traffic types: In this thesis we only considered an internal collision between two different VoIP traffics. What is the performance of the internal collision mechanism? How well does it perform when all traffic classes are used with background, best-effort, voice and video traffic?

The quality requirements of VoIP and the characteristics of the WLAN protocol interact in surprising ways. This thesis has explored some of the those interactions. Nevertheless there is considerable future work to be carried out as newer applications emerge and the series of IEEE 802.11 protocols is developed further.

Bibliography

[1] D J Leith, P Clifford, D Malone, and A Ng. TCP fairness in. 802.11e WLANs. *IEEE Communications Letters*, 9(11):964–966, 2005.

[2] Ian Dangerfield, David Malone, and Douglas J Leith. Understanding 802.11e Voice Behaviour via Testbed Measurements and Modeling. *5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops (WiOpt'07)*, pages 1–6, April 2007.

[3] ITU-T Recommendation P.59: Telephone Transmission Quality Objective Measureing Apparatus, Articifical Conversational Speech, 1993.

[4] IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, 2007.

[5] Cisco Systems. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2010-2015, February 2011.

[6] ETSI. Broadband Radio Access Networks (BRAN); HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1; Functional specification. EN300652, July 1998.

[7] ETSI. Broadband Radio Access Networks (BRAN);HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1;Conformance testing specification;Part 1: Radio type approval and Radio Frequency (RF) conformance test specification. ETS300836-1, May 1998.

[8] ETSI. Broadband Radio Access Networks (BRAN);HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1;Conformance testing specification;Part 2: Protocol Implementation Conformance Statement (PICS) proforma specification. ETS300836-2, May 1998.

[9] ETSI. Broadband Radio Access Networks (BRAN);HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1;Conformance testing specification;Part 3: Test Suite Structure and Test Purposes (TSS&TP) specification. ETS300836-3, May 2005.

[10] ETSI. Broadband Radio Access Networks (BRAN);HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1;Conformance testing specification;Part 4: Abstract Test Suite (ATS) specification. ETS300836-4, May 1998.

[11] IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, High-speed Physical Layer in the 5 GHz Band, 1999.

[12] IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, High-speed Physical Layer in the 2.4 GHz Band, 1999.

[13] IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, 1999.

[14] IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, 2005.

[15] Market Clarity. The Aussie VoIP list. http://www.marketclarity.com.au/voip/.

[16] Alcatel Lucent. Spectacular cost reduction - Thanks to State-of-the-Art VoIP solution to Agrana, April 2009.

[17] Kewin O Stoeckigt and Hai L Vu. VoIP Capacity Analysis in IEEE 802.11 WLAN. *In Proceedings of the 34th IEEE Conference on Local Computer Networks (LCN'09)*, pages 116–123, 2009.

[18] ITU-T Recommendation P.800: Series P: Telephone Transmission Quality, Methods for objective and subjective assesment of quality, Methods for subjective determination of transmission quality, 1996.

[19] ITU-T Recommendation G.107: The E-model, a computational model for use in transmission planning, 2005.

[20] R Bruno, M Conti, and E Gregori. Throughput Analysis of UDP and TCP Flows in IEEE 802.11b WLANs: A Simple Model and Its Validation. *Workshop on Techniques, Methodologies and Tools for Performance Evaluation of Complex Systems*, pages 54–63, 2005.

[21] Boris Bellalta, Michela Meo, and Miquel Oliver. Comprehensive analytical models to evaluate the TCP Performance in 802.11 WLAN. *Lecture Notes in Computer Science*, pages 37–48, 2006.

[22] IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, Amendment 5: Enhancements for higher thoughput, 2009.

[23] J Rosenberg, H Schulzrinne, G Camarillo, A Jonston, J Peterson, R Sparks, M Handley, and E Schooler. SIP: Session Initiation Protocol, June 2002.

[24] ITU-T Recommendation H.323, Series H: Audiovisual and Multimedia Systems; Infrastructure of audiovosual services - Systems and terminal equipment for audioviosual services; Packet-based multimedia communications systems, 2006.

[25] David Malone, Peter Clifford, and D.J. Leith. On buffer sizing for voice in 802.11 WLANs. *IEEE Communications Letters*, 10(10):701–703, October 2006.

[26] Sri Harsha, Anurag Kumar, and Vinod Sharma. Analytical model for an IEEE 802.11 WLAN using DCF with two types of VoIP calls. *National Communications Conference (NCC) New Delhi*, 2006.

[27] Sangho Shin and Henning Schulzrinne. Experimental Measurement of the Capacity for VoIP Traffic in IEEE 802.11 WLANs. *26th IEEE International Conference on Computer Communications (INFOCOM'07)*, pages 2018–2026, May 2007.

[28] Australian Communication and Media Authority (ACMA). The Australian VoIP Market - The supply and take-up of VoIP in Australia. 2008.

[29] Australian Communication and Media Authority (ACMA). Changes in the Australian VoIP market, December 2009.

[30] TeleGeography. TeleGeography Report & Database. http://bit.ly/tuX18H.

[31] TechReaders. Skype Enjoys 12% Market Share of International Calls. http://bit.ly/t5dVwr.

[32] TeleGeography. Skype's share of the international long-distance pie on the increase. http://bit.ly/vpWNAu, March 2009.

[33] TeleGeography. International long-distance slupms, while Skype soars. http://bit.ly/tSA2wn, January 2011.

[34] Richard Kuhn, Thomas J Walsh, and Steffen Fries. Security Considerations for Voice over IP Systems. Technical Report 800-58, Computer Security Division - National Institute of Standards and Technology, 2005.

[35] Angelos D Keromytis. A Survey of Voice over IP Security Research. In *Proceedings of the 5th International Conference on Information Systems Security (ICISS'09)*, pages 1–17, 2009.

[36] Anthony Plewes. The biggest VoIP security threats - and how to stop them. *Silicon.com Whitepaper*, 2007.

[37] Simson L Garfinkel. VoIP and Skype Security. http://www.cs.columbia.edu/ salman/skype/, 2005.

[38] Phillippa Biggs. VoIP: Current Trends and Future Evolution. In *Telecom Evolution Business Forum*, October 2006.

[39] Ostermann Research. Enterprise VoIP Market Trends, 2009 - 2012. http://www.ostermanresearch.com, 2009.

[40] James F Kurose and Keith W Ross. *Computer Networking - A top down approach featuring the Internet*. Addison Wesley, 2003.

[41] Xuemin Shen, Heather Yu, John Buford, and Musalin Akon. *Handbook for Peer-To-Peer Networking*. Springer, 2010.

[42] Quang Hieu Vu, Mihai Lupu, and Beng Chin Ooi. *Peer-to-peer Computing*. Springer, 2010.

[43] Saikat Guha, Neil Daswani, and Ravi Jain. An experimental study of Skype Peer-to-peer VoIP system. *Proceedings of 5th International Workshop on Peer-To-Peer Systems (IPTPS'06)*, 2006.

[44] Haiyong Xie and Yang Richard Yang. A measurement-based study of the Skype peer-to-peer VoIP performance. *Proceedings of the 6th International Workshop on Peer-to-Peer Systems (IPTPS'07)*, 2007.

[45] Salman A. Baset and Henning Schulzrinne. An analysis of the Skype peer-to-peer VoIP system. *IEEE International Conference on Computer Communications (INFOCOM'06)*, 2006.

[46] Scott Bradner, Lawrence Conroy, and Kazunori Fujiwara. The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM). RFC6116, March 2011.

[47] Michael Mealling. Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database. RFC3403, October 2002.

[48] Kewin Stoeckigt. Videokonferenzen aus sicheren Netzen. Universitaet Goettingen, B.Sc. thesis, 2002.

[49] Salaman A Baset and Henning Schulzrinne. An analysis of the Skype peer-to-peer Internet telephony protocol. Technical Report 39, Columbia University, September 2004.

[50] Dario Bonfiglio, Marco Mellia, Michela Meo, and Dario Rossi. Detailed analysis of Skype traffic. *IEEE Transactions on Multimedia*, 11(1):117–128, January 2009.

[51] Philip Branch, Amil Heyde, and Grenville Armitage. Rapid identificagion of Skype traffic. In *19th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'09)*, 2009.

[52] Peter Saint-Andre. Extensible Messaging and Presence Protocol. RFC3920, 2004.

[53] ETSI. Digital cellular telecommunications system (Phase 2+) (GSM); Full rate speech; Transcoding. Technical report, 2000.

[54] ITU-T Recommendation G.711: General Aspects of Digitial Transmission Systems, Terminal Equipments, Pulse Code Modulation (PCM) of Voice Frequency, Appendix II: A comfort noise payload definition for ITU-T G.711 use in packet-based multimedia communication, 2000.

[55] ITU-T Recommendation G.729: Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP), 2007.

[56] ITU-T Recommendation G.722.1: Low-complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss, 2005.

[57] Henning Schulzrinne, S Casner, R Frederick, and Van Jacobson. Real-time transport protocol. RFC3550, 2003.

[58] Skype Technologies S.A. SILK - Super Wideband Audio Codec.

[59] A.P. Markopoulou, F.A. Tobagi, and M.J. Karam. Assessment of VoIP quality over Internet backbones. *Proceedings of the 25th Annual Joint Conference of the IEEE Computer and Communications Societies*, 1:150–159, 2002.

[60] A.D. Clark. Modeling the Effects of Burst Packet Loss and Recency on Subjective Voice Quality. In *IPTel 2001*.

[61] Choon Shim, Liehue Xie, Bryan Zhang, and C J Sloane. How Delay and Packet loss impact voice quality in VoIP. Technical report, Quova Inc., 2003.

[62] Mohammad Ilyas and Syed A Ahson. *VoIP Handbook*. CRC Press Inc., December 2008.

[63] Kyungtae Kim and Young-June Choi. Performance comparison of various VoIP codecs in wireless environments. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication (ICUIMC '11)*, volume 89, pages 1–10, New York, NY, USA, 2011.

[64] F Anjum, M Elaoud, D Famolari, A Ghosh, R Vaidyanathan, A Dutta, P Agrawal, T Kodama, and Y Katsube. Voice performance in WLAN networks - an experimental study. *IEEE Global Telecommunications Conference (GLOBECOM-2003)*, 6:3504–3508, December 2003.

[65] K Medepalli, P Gopalakrishnan, D Famolari, and T Kodama. Voice capacity of IEEE 802.11b, 802.11a and 802.11g wireless LANs. *IEEE Global Telecommunications Conference (GLOBECOM'04)*, 3:1549–1553, 2004.

[66] Kei Igarashi, Akira Yamada, and Tomoyuki Ohya. Capacity improvement of wireless LAN VoIP using distributed transmission scheduling. In *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07)*, pages 1–5, 2007.

[67] Wenyu Jiang and Henning Schulzrinne. Comparisons of FEC and codec robustness on VoIP quality and bandwidth efficiency. In *Proceedings of the 7th International Conference on Networking (ICN'02)*, pages 1–12, 2002.

[68] J.; Brada M.; Novak L Becvar Z.; Zelenka. Comparison of Common PLC Methods Used in VoIP Networks. *Proceedings of the 6th EURASIP Conference focused on Seech and Image processing, multimedia communication and systems*, pages 389–392, 2007.

[69] J H James, Bing Chen, and L Garrison. Implementing VoIP: a voice transmission performance progress report. *IEEE Communications Magazine*, 42(7):36–41, July 2004.

[70] R V Cox, D Malah, and D Kapilow. Improving upon toll quality speech for VoIP. *Conference Record of the 38 Asilomar Conference on Signals, Systems and Computers*, 1:405 – 409, 2004.

[71] Christian Hoene, Berthold Rathke, and Adam Wolisz. On the importance of a VoIP packet. *1st International Speech Communication Association ISCA Tutorial and Research Workshop on Auditory Quality of Systems*, 2003.

[72] M Raazvi Doomun. Adaptive packet loss concealment mechanism for wireless voice over IP. *Innovative Algorithms and Techniques in Automation, Industrial Electronics and Telecommunications*, pages 375–377, 2007.

[73] D P Hole and F A Tobagi. Capacity of an IEEE 802.11b wireless LAN supporting VoIP. *IEEE International Conference on Communications (ICC'04)*, 1:196–201, June 2004.

[74] Chen-Chi Wu, Kuan-Ta Chen, Yu-Chun Chang, and Chin-Laung Lei. An empirical evaluation of VoIP blayout buffer dimensioning in Skype, Google Talk, and MSN Messenger. *19th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'09)*, pages 97–102, 2009.

[75] John Q. Walker. Assessing VoIP call quality using the E-model. Technical report, netiQ, 2002.

[76] Cisco Systems. Understanding delay in packet voice networks. http://goo.gl/k2Vtu.

[77] Alan Clark. Voice quality measurement: Understanding VoIP. EE Times Asia, 2005.

[78] ITU-T Recommendation P.862: Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, 2001.

[79] ITU-T Recommendation G.113: Provisional planning values for the equipment impairment factor Ie and packet-loss robust, 2007.

[80] ITU-T Recommendation P.64: Determination of sensitivity/frequency characteristics of local telephone systems, 1999.

[81] Harvard Sentences. *IEEE Transactions on Audio and Electroacoustics*, 17:227–246, 1996.

[82] Ian. F Akyldiz and Xudong Wang. Wireless mesh networks: a survey. *Computer Networks*, 47(4):445–487, March 2005.

[83] Douglas E. Comer. *Computer Networks and Internets*. Pearson Education Inc., 2009.

[84] Matthew Gast. *802.11 Wireless Networks: The Definitive Guide*. O'Reilly, January 2005.

[85] Juha Heiskala and John Terry. *OFDN Wireless LANs: A theoretical and Practical Guide*. Sams, 2001.

[86] Jost Weinmiller, Hagen Woesner, Jean-pierre Ebert, and Adam Wolisz. Analyzing the RTS/CTS Mechanism in the DFWMAC Media Access Protocol for Wireless LANs, 1995.

[87] Kaixin Xu, M Gerla, and Sang Bae. How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks. In *IEEE Global Telecommunications Conference (GLOBECOM'02)*, volume 1, pages 72 – 76, 2002.

[88] Takehiro Kawata. Using dynamic PCF to improve the capacity for VoIP traffic. In *IEEE Wireless Communications and Networking Conference (WCNC'05)*, volume 3, pages 1589–1595, 2005.

[89] Giuseppe Bianchi, Tor Vergata, Ilenia Tinnirello, and Luca Scalia. Understanding 802.11e Contention-based prioritization mechansims and their coexistence with legacy 802.11 stations. *IEEE Network*, 19(4):28–34, 2005.

[90] Sri Harsha, S V R Anaud, Anurag Kumar, and Vinod Sharma. An Analytical model for Capacity Evaluation of VoIP on HCCA and TCP File Transfers over EDCA in an IEEE 802.11e WLAN. *Lecture Notes in Computer Science*, 4308/2006:245–256, 2006.

[91] Claudio Cicconetti, Luciano Lenzini, Enzo Mingozzi, and Giovanni Stea. Design and performance analysis of the Real-Time HCCA scheduler for IEEE 802.11e WLANs. *Computer Networks*, 51(9):2311–2325, 2007.

[92] A. Gondran, O. Baala, A. Caminada, and H. Mabed. Interference Management in IEEE 802.11 Frequency Assignment. In *IEEE Vehicular Technology Conference (VTC'08)*, pages 2238–2242. IEEE, May 2008.

[93] Vijay Subramanian, K. K. Ramakrishnan, Shiv Kalyanaraman, and Lusheng Ji. Impact of interference and capture effects in 802.11 wireless networks on TCP. In *Proceedings of the 2nd international workshop on Wireless traffic measurements and modeling (WitMeMo '06)*, New York, New York, USA, 2006. ACM Press.

[94] C. McKay and F. Masuda. Empirical studies of wireless VoIP speech quality in the presence of Bluetooth interference. In *IEEE Symposium on Electromagnetic Compatibility*, volume 1, pages 269–272, 2003.

[95] N. Golmie, R. E. Van Dyck, and A. Soltanian. Interference of bluetooth and IEEE 802.11. In *Proceedings of the 4th ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems (MSWIM '01)*, pages 11–18, New York, New York, USA, 2001. ACM Press.

[96] N Golmie, R E Van Dyck, A Soltanian, A Tonnerre, and O Rébala. Interference Evaluation of Bluetooth and IEEE 802.11b Systems. *Wireless Networks*, 9(3):201–211, 2003.

[97] A.H. Nour-Eldin, H. Tolba, and D. O'Shaughnessy. Automatic recognition of Bluetooth speech in 802.11 interference and the effectiveness of insertion-based compensation techniques. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04)*, volume 1, pages I–1033–6.

[98] O. Awoniyi and F.A. Tobagi. Effect of fading on the Performance of VOIP in IEEE 802.11 a WLANs. *IEEE International Conference on Communications*, 6:3712–3717, 2004.

[99] T.T. Kwon, M. Gerla, and S. Das. Mobility management for VoIP service: Mobile IP vs. SIP. *IEEE Wireless Communications*, 9(5):66–75, October 2002.

[100] I. Samprakou, C. Bouras, and T. Karoubalis. Fast IP Handoff Support for VoIP and Multimedia Applications in 802.11 WLANs. In *6th IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM'05)*, pages 332–337, 2005.

[101] Arlindo Da Conceicao, Jin Li, Dinei Florencio, and Fabio Kon. Is IEEE 802.11 ready for VoIP? In *IEEE Workshop on Multimedia Signal Processing*, pages 108–113. IEEE, October 2006.

[102] Nidhi Hedge, Alexandre Proutiere, and James Roberts. Evaluating the voice capacity of 802.11 WLAN under distributed control. *The 14th IEEE workshop on Local and Metropolitan area networks (LANMAN'05)*, pages 6–11, 2005.

[103] Lin X Cai, Xuemin Shen, Jon W Mark, Lin Cai, and Yang Xiao. Voice capacity analysis of WLAN with unbalanced traffic. *IEEE Transactions on Vehicular Technology*, pages 752–761, 2006.

[104] S Harsha. An analytical model for performance evaluation of multimedia applications over EDCA in an IEEE 802.11e WLAN. *Wireless Networks*, 16(2):367–385, 2010.

[105] Byung Joon Oh and Chang Wen Chen. Analysis of Retry Limit for Supporting VoIP in IEEE 802.11e EDCA WLANs. *16th International Conference on Computer Communications and Networks (ICCCN'07)*, pages 464–469, August 2007.

[106] S Garg and M Kappes. Can I add a VoIP call? In *IEEE International Conference on Communications (ICC '03)*, volume 2, pages 779–783. IEEE, May 2003.

[107] Lin X Cai, Xinhua Ling, Xuemin Shen, Jon W Mark, and Lin Cai. Supporting voice and video applications over IEEE 802.11n WLANs. *Wireless Networks*, pages 443–454, 2009.

[108] M Elaoud, D Famolari, and A Ghosh. Experimental VoIP capacity measurements for 802.11b WLANs. *Second IEEE Consumer Communications and Networking Conference (CCNC'05)*, pages 272–277, January 2005.

[109] H Schulzrinne. Measurement and Analysis of the VoIP Capacity in IEEE 802.11 WLAN. *IEEE Transactions on Mobile Computing*, 8(9):1265–1279, September 2009.

[110] Philipp Hofmann, Chunlei An, Luis Loyola, and Imad Aad. Analysis of UDP, TCP and Voice Performance in IEEE 802.11b Multihop Netwrks. In *Proceedings of the 13th European Wireless Conference*, pages 1–4, 2007.

[111] Carlos Henrique R de Oliveira and Marcos Antonio de Siqueira. Simultaneous VoIP Calls Capacity over an 802.11 Ad hoc network. *Proceedings of the 2006 International Conference on Wireless Networks (ICWN'06)*, 2006.

[112] Lizzie Edmea Narvaez Diaz and Jesus Arturo Perez Diaz. A Model for Designing WLAN's 802.11 for VoIP. *Electronics, Robotics and Automotive Mechanics Conference*, 2:110–115, 2006.

[113] A Trad, F Munir, and H Afifi. Capacity evaluation of VoIP in IEEE 802.11e WLAN environment. In *3rd IEEE Consumer Communications and Networking Conference (CCNC'06)*, volume 2, pages 828–832, 2006.

[114] Dongyan Chen, Sachin Garg, Martin Kappes, and Kishor S Trivedi. Supporting VBR VoIP Traffic in IEEE 802.11 WLAN in PCF mode. Technical Report 26, Avaya Labs, 2002.

[115] A Rumín. VoIP over WLAN 802.11 b simulations for infrastructure and ad-hoc networks. *Proceedings of the London Communications Symposium (LCS'06)*, January 2006.

[116] J Barcelo, B Bellalta, C Cano, and A Sfairopoulou. VoIP Packet Delay in Single-Hop Ad-Hoc IEEE 802.11 Networks. *Fifth Annual Conference on Wireless on Demand Network Systems and Services (WONS-2008)*, pages 77–80, January 2008.

[117] S Armenia, L Galluccio, A Leonardi, and S Palazzo. Transmission of VoIP Traffic in Multihop Ad Hoc IEEE 802.11b Networks: Experimental Results. *First International Conference on Wireless Internet (WICON'05)*, pages 148–155, 2005.

[118] Moshe Zukerman. Introduction to Queueing Theory and Stochastic Teletraffic models. http://www.ee.cityu.edu.hk/˜zukerman/classnotes.pdf.

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

[120] S Harsha, A Kumar, and V Sharma. An Analytical Model for the Capacity Estimation of Combined VoIP and TCP File Transfers over EDCA in an IEEE 802.11e WLAN. *14th IEEE International Workshop on Quality of Service (IWQoS'06)*, pages 178–187, June 2006.

[121] J. del Prado Pavon and P. Wienert. Optimal packing of VoIP calls in an IEEE 802.11a/e WLAN in the presence of QoS constraints and channel errors. In *IEEE Global Telecommunications Conference (GLOBECOM '04)*, volume 5, pages 2974–2980. IEEE, 2004.

[122] Deyun Gao, Jianfei Cai, and Chang Wen Chen. WLC19-6: Capacity Analysis of Supporting VoIP in IEEE 802.11e EDCA WLANs. *IEEE Global Telecommunications Conference (GLOBECOM'06)*, pages 1–5, November 2006.

[123] Deyun Gao, Jianfei Cai, Chuan Heng Foh, Chiew-Tong Lau, and King Ngi Ngan. Improving WLAN VoIP capacity through service differentiation. *IEEE Transactions on Vehicular Technology*, 57:465–474, January 2008.

[124] J. Jakubiak and Y. Koucheryavy. Precise Delay Analysis for IEEE 802.11 Legacy Ad-Hoc Networks. *IEEE 65th Vehicular Technology Conference (VTC-2007)*, pages 2956–2960, April 2007.

[125] Xinhua Ling, Yu Cheng, Xuemin Shen, and J W Mark. Voice capacity analysis of WLANS with channel access prioritizing mechanisms. *IEEE Communications Magazine*, 46(1):82–89, January 2008.

[126] Ping Wang, Hai Jiang, and Weihua Zhuang. A new MAC scheme supporting voice/data traffic in wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 7(12):1491–1503, December 2008.

[127] Hung yu Wei, Kyungtae Kim, A. Kashyap, and S. Ganguly. On admission of VoIP calls over wireless mesh network. *IEEE International Conference on Communications (ICC'06)*, 5:1990–1995, January 2006.

[128] Wei Wang, Soung Chang Liew, and Victor O K Li. Solutions to performance problems in VoIP over a 802.11 wireless LAN. *IEEE Transactions on Vehicular Technology*, pages 366–384, 2005.

[129] Zhuo Chen, Lingyun Wang, Feng Zhang, Xinbing Wang, and Wen Chen. VoIP over WLANs by Adapting Transmitting Interval and Call Admission Control. In *IEEE International Conference on Communications (ICC'08)*, pages 3242–3246, 2008.

[130] P Clifford, K Duffy, D.J. Leith, and D Malone. On Improving Voice Capacity in 802.11 Infrastructure Networks. *International Conference on Wireless Networks, Communications and Mobile Computing*, 1:214–219, June 2005.

[131] Yeonsik Jeong, Kakumanu Sandeep, Cheng-Lin Tsao, and Raghupathy Sivakumar. Improving VoIP call capacity over IEEE 802.11 networks. *4th International Conference on Broadband Communications, Networks and Systems (BROADNETS'07)*, pages 670–679, 2007.

[132] Wei Wang, S.C. Liew, Qixiang Pang, and V.O.K. Li. A multiplex-multicast scheme that improves system capacity of voice over-IP on wireless LAN by 100%. *Proceedings of the 9th International Symposium on Computers And Communications (ISCC'04)*, 1:472–477, 2004.

[133] Daniel E Lucani, Renny E Badra, and Carlos M Bianchi. Increasing VoIP capacity on WiFi networks through the use of the FAIR algorithm for

MAC. *IEEE 65th Vehicular Technology Conference (VTC-2007)*, pages 1684 –1688, 2007.

[134] J Freitag, N L S da Fonseca, and J F de Rezende. Tuning of 802.11e network parameters. *IEEE Communications Letters*, 10:611–613, 2006.

[135] J Freitag, N L S da Fonseca, and J Ferreira de Rezende. A Tuning Mechanism for 802.11e Access Parameters for Coping with Delay Asymmetry. *IEEE International Conference on Communications (ICC'06)*, 2:540–545, June 2006.

[136] Ping Wang, Hai Jiang, and Weihua Zhuang. Capacity improvement and analysis for voice/data traffic of WLANs. *IEEE Transactions on Wireless Communications*, 6(4):1530–1541, April 2007.

[137] M. Irfan Anis, M. Zamin Khan, Jahanzeb Inam, and Rana Azmatullah. Comparison between the IEEE 802.11x standards of VoIP using the CBR and VBR voice schemes. *IEEE Radio and Wireless Symposium*, pages 207–210, January 2008.

[138] Seoungyoul Oh, Jongmin Shin, Dongho Kwak, and Cheeha Kim. A Novel Call Admission Control Scheme for the IEEE 802.11e EDCA. In *10th International Conference on Advanced Communication Technology (ICACT'08)*, volume 3, pages 1832–1835, February 2008.

[139] Yi Qian, R.Q.-Y. Hu, and Hsiao-Hwa Chen. A call admission control framework for voice over WLANs. *IEEE Wireless Communications*, 13(1):44–50, February 2006.

[140] Hongqiang Zhai, Xiang Chen, and Yuguang Fang. A call admission and rate control scheme for multimedia support over IEEE 802.11 wireless LANs. *Wireless Networks*, 12(4):451–463, July 2006.

[141] Cisco Systems. VoIP Call Admission Control. pages 1–50, 2001.

[142] Samrat Ganguly and Sudeept Bhatnagar. *VoIP: wireless, P2P and New Enterprise Voice over IP*. John Willey & Sons, Ltd, 2008.

[143] Alessandro Andreadis, Giuliano Benelli, and Riccardo Zambon. An admission control algorithm for QoS provisioning in IEEE 802.11e EDCA. *3rd International Symposium on Wireless Pervasive Computing*, pages 298–302, May 2008.

[144] Huanxu Pan and Satoru Yamano. An analysis of the channel utilization for call admission control in voice over WLAN. In *Proceedings of the 2006 international conference on Wireless communications and mobile computing (IWCMC '06)*, pages 1229–1234, New York, NY, USA, 2006. ACM.

[145] Miroslaw Narbutt and Mark Davis. Experimental tuning of AIFSN and CWmin parameters to prioritize voice over data transmission in 802.11e WLAN networks. In *Proceedings of the 2007 international conference on Wireless communications and mobile computing (IWCMC'07)*, pages 140–145, 2007.

[146] C Casetti and C-F Chiasserini. Improving fairness and throughput for voice traffic in 802.11e EDCA. *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'04)*, 1:525–530, September 2004.

[147] H Tode and K Murakami. Friendly Coexistence of Voice and Data Traffic in IEEE 802.11 WLANs. *IEEE Transactions on Consumer Electronics*, 52(2):347–354, May 2006.

[148] Amogh Dhamdhere and Constantine Dovrolis. Open issues in router buffer sizing. *ACM SIGCOMM Computer Communication Review*, 36(1):87, January 2006.

[149] D. Barman, G. Smaragdakis, and I. Matta. The effect of router buffer size on highspeed TCP performance. *IEEE Global Telecommunications Conference (GLOBECOM'04)*, 3:1617–1621, 2004.

[150] Guido Appenzeller, Isaac Keslassy, and Nick McKeown. Sizing router buffers. *ACM SIGCOMM Computer Communication Review*, 34(4):281, October 2004.

[151] Damon Wischik and Nick McKeown. Part I: Buffer Sizes for core routers. *ACM SIGCOMM Computer Communication Review*, 35(3):75, July 2005.

[152] Neda Beheshti, Yashar Ganjali, Monia Ghobadi, Nick McKeown, and Geoff Salmon. Experimental study of router buffer sizing. *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement conference (IMC '08)*, pages 197–210, 2008.

[153] S. Iyer, R.R. Kompella, and N. McKeown. Designing Packet Buffers for Router Linecards. *IEEE/ACM Transactions on Networking*, 16(3):705–717, June 2008.

[154] Yashar Ganjali and Nick McKeown. Update on buffer sizing in internet routers. *ACM SIGCOMM Computer Communication Review*, 36(5):67–70, October 2006.

[155] Jeffrey Semke, Jamshid Mahdavi, and Matthew Mathis. Automatic TCP buffer tuning. *ACM SIGCOMM Computer Communication Review*, 28(4):315–323, October 1998.

[156] Arun Vishwanath, Vijay Sivaraman, and Marina Thottan. Perspectives on router buffer sizing. *ACM SIGCOMM Computer Communication Review*, 39(2):34, March 2009.

[157] S. Gorinsky, A. Kantawala, and J. Turner. Link Buffer Sizing: A New Look at the Old Problem. *10th IEEE Symposium on Computers and Communications (ISCC'05)*, pages 507–514, 2005.

[158] Tianji Li and D J Leith. Buffer Sizing for TCP Flows in 802.11e WLANs. *IEEE Communications Letters*, 12(3):216–218, March 2008.

[159] Andrew S. Tanenbaum. *Computer Networks*. Wetherall, 2003.

[160] Koen Vos, Soeren Skak Jensen, and Karsten Vandborg Soerensen. SILK Speec Codec. http://tools.ietf.org/html/draft-vos-silk-02, September 2010.

[161] Greg Herlein, Simon Morlat, Jean-Marc Valin, and Roger Hardiman. SPEEX - A free codec for free speec. http://www.speex.org, April 2005.

[162] Anna Sfairopoulou, Carlos Macian, and Boris Bellalta. VoIP codec adaptation algorithm in multirate 802.11 WLANs: Distributed vs. centralized performance comparison. *Lecture Notes in Computer Science*, 4606:52–61, 2007.

[163] Boris Bellalta, Michela Meo, and Miquel Oliver. A BEB-based admission control for VoIP calls in WLAN with coexisting elastic TCP flows. *Lecture Notes in Computer Science*, 4003:130–141, 2006.

[164] T Bu, Y Liu, and D Towsley. On the TCP-Friendliness of VoIP Traffic. *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM'06)*, pages 1–12, March 2006.

[165] C. Brouzioutis, V. Vitsas, and P. Chatzimisios. Studying the Impact of Data Traffic on Voice Capacity in IEEE 802.11 WLANs. *IEEE International Conference on Communications (ICC)*, pages 1–6, April 2010.

[166] Jeongyun Yu, Sunghyun Choi, and Daji Qiao. TCP dynamics over IEEE 802.11E WLANs: Modeling and throughput enhancement. In *4th International Conference on Broadband Communications, Networks and Systems (BROADNETS '07)*, pages 66–75. IEEE, 2007.

[167] Raffaele Bruno, Marco Conti, and Enrico Gregori. Performance modelling and measurements of TCP transfer throughput in 802.11-based WLAN. *Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems (MSWiM '06)*, pages 4–11, 2006.

[168] Wen-Kuang Kuo. Supporting multimedia streaming and best-effort data transmission over IEEE 802.11e EDCA. *International Journal of Network Management*, 18(3):209–228, May 2008.

[169] George Kuriakose, Sri Harsha, Anurag Kumar, and Vinod Sharma. Analytical models for capacity estimation of IEEE 802.11 WLANs using DCF for internet applications. *Wireless Networks*.

[170] Jeonggyun Yu, Sunghyun Choi, and Jaehwan Lee. Enhancement of VolP over IEEE 802.11 WLAN via dual queue strategy. In *IEEE International Conference on Communications (ICC '04)*, volume 6, pages 3706 – 3711, 2004.

[171] Wen-Tzu Chen. An effective medium contention method to improve the performance of IEEE 802.11. *Wireless Networks*, 14:769–776, 2007.

[172] S Garg and M Kappes. Admission control for VoIP traffic in IEEE 802.11 networks. In *IEEE Global Telecommunications Conference (GLOBECOM'03)*, volume 6, pages 3514 – 3518, 2003.

[173] Syed A Ahson. *VoIP Handbook: Applications, Techniques, Reliability, and Security*. 2008.

[174] M Karam and Fouad A. Tobagi. Analysis of delay and delay jitter of voice traffic in the Internet. *Computer Networks*, 40(6):711–726, December 2002.

[175] O Tickoo and B Sikdar. Queueing analysis and delay mitigation in IEEE 802.11 random access MAC based wireless networks. *23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, 2:1404–1413, March 2004.

[176] Leonard Kleinrock. Queueing Systems, Volume 1: Theory. *Wiley-Interscience*, 1975.

[177] The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns/.

[178] Sven Wiethoelter, Marc Emmelmann, Christian Hoene, and Adam Wolisz. TKN EDCA Model for ns-2. Technical report, 2006.

[179] Sven Wiethoelter and Christian Hoene. Design and Verification of an IEEE 802.11e EDCF Simulation Model in ns-2.26. Technical report, TU Berlin, 2003.

[180] Sven Wiethoelter, Christian Hoene, and Adam Wolisz. Perceptual Quality of Internet Telephony over IEEE 802.11e Support enhanced DCF in contention free burst. Technical report, TU Berlin, 2004.

[181] WiFi Alliance. Wireless Multimedia Extensions. http://www.wi-fi.org/knowledge_center/wmm.

[182] TinyBSD. http://www.freebsd.org.

[183] Ken Duffy and Ayalvadi Ganesh. Modeling the Impact of Buffering on 802.11. *IEEE Communications Letters*, 11(2):219–221, February 2007.

[184] Lawrence Stewart, David A Hayes, Grenville Armitage, Michael Welzl, and Andreas Petlund. Multimedia-unfriendly TCP congestion control and home gateway queue management. In *Proceedings of the 2nd annual ACM conference on Multimedia systems (MMSys'11)*, pages 35–44, New York, NY, USA, 2011.

[185] ITU-T Recommendation G.114: Transmission Systems and Media, Digital Systems and Networks; One-way transmission time, 2003.

[186] Kewin O Stoeckigt and Hai L Vu. VoIP Capacity - Analysis, Improvements and Limits in IEEE 802.11 Wireless LAN. *IEEE Transactions on Vehicular Technology*, 59(9):4553–4563, 2010.

[187] M. Allman, V. Paxson, and E. Blanton. TCP Congestion Control. *RFC5681*, 2009.

[188] Van Jacobson. Congestion avoidance and control. *ACM SIGCOMM Computer Communication Review*, 18(4):314–329, August 1988.

[189] W. Richard Stevens. *TCP/IP Illustrated - Volume 1*. Addison Wesley, 26 edition, 2005.

[190] L Bononi, M Conti, and E Gregori. Runtime optimization of IEEE 802.11 wireless LANs performance. *IEEE Transactions on Parallel and Distributed Systems*, 15(1):66–80, January 2004.

Appendices

Appendix A

priority.tcl

The following shows the default setting of the `priority.tcl` file of the EDCA extension by the TU-Berlin [179, 180].

```
[...]

# 802.11b parameters (default EDCA parameter set),
# aCWmin=31, aCWmax=1023
proc priority { ifq_name } {
   upvar $ifq_name ifq

    # parameters for Queue 0
    $ifq Prio 0 PF 2
    $ifq Prio 0 AIFS 2
    $ifq Prio 0 CW_MIN 7           ;# (aCWmin+1)/4 - 1
    $ifq Prio 0 CW_MAX 15          ;# (aCWmin+1)/2 - 1
    $ifq Prio 0 TXOPLimit 0.003264

    #parameters for Queue 1
    $ifq Prio 1 PF 2
    $ifq Prio 1 AIFS 2
    $ifq Prio 1 CW_MIN 15          ;# (aCWmin+1)/2 - 1
    $ifq Prio 1 CW_MAX 31          ;# aCWmin
    $ifq Prio 1 TXOPLimit 0.006016

    #parameters for Queue 2
    $ifq Prio 2 PF 2
```

```
    $ifq Prio 2 AIFS 3
    $ifq Prio 2 CW_MIN 31          ;# aCWmin
    $ifq Prio 2 CW_MAX 1023        ;# aCWmax
    $ifq Prio 2 TXOPLimit 0

    #parameters for Queue 3
    $ifq Prio 3 PF 2
    $ifq Prio 3 AIFS 7
    $ifq Prio 3 CW_MIN 31          ;# aCWmin
    $ifq Prio 3 CW_MAX 1023        ;# aCWmax
    $ifq Prio 3 TXOPLimit 0
}
```

Wired/Wireless node VoIP codec configuration

The following shows the configuration of a VoIP node (wired/wireless). Note that the parameters for `$byte` and `$ival` was set according to the used voice codec, e.g. `$byte = 10` and `$ival = 0.01` for a G.729 voice codec with a 10 ms sampling rate. Also note that the value for `$prio_` was adjusted for the AP or the wireless nodes.

```
[...]

set vnode [new Agent/UDP]
$vnode set class_ 0
$vnode set prio_ 1
$ns attach-agent $WN $vnode
set vnode_traffic [new Application/Traffic/CBR]
$vnode_traffic set packetsize_ $bytes
$vnode_traffic set interval_ $ival

[...]
```