

Voice Capacity in Opportunistic Spectrum Access  
Networks with Friendly Scheduling

VOICE CAPACITY IN OPPORTUNISTIC SPECTRUM ACCESS  
NETWORKS WITH FRIENDLY SCHEDULING

BY

HANAN S. HASSANEIN, M.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

© Copyright by Hanan S. Hassanein, September 2016

All Rights Reserved

Doctor of Philosophy (2016)  
(Electrical & Computer Engineering)

McMaster University  
Hamilton, Ontario, Canada

TITLE: Voice Capacity in Opportunistic Spectrum Access Networks  
with Friendly Scheduling

AUTHOR: Hanan S. Hassanein  
M.Sc., (Computer Engineering)  
Cairo University, Giza, Egypt

SUPERVISOR: Dr. Terence D. Todd

NUMBER OF PAGES: xx, 165

# Abstract

Radio spectrum has become increasingly scarce due to the proliferation of new wireless communication services. This problem has been exacerbated by fixed bandwidth licensing policies that often lead to spectral underutilization. Cognitive radio networks (CRN) can address this issue using flexible spectrum management that permits unlicensed (secondary) users to access the licensed spectrum. Supporting real-time quality-of-service (QoS) in CRNs however, is very challenging, due to the random spectrum availability induced by the licensed (primary) user activity. This thesis considers the problem of real-time voice transmission in CRNs with an emphasis on secondary network “friendliness”. Friendliness is measured by the secondary real-time voice capacity, defined as the number of connections that can be supported, subject to typical QoS constraints.

The constant bit rate (CBR) air interface case is first assumed. An offline scheduler that maximizes friendliness is derived using an integer linear program (ILP) that can be solved using a minimum cost flow graph construction. Two online primary scheduling algorithms are then introduced. The first algorithm is based on shaping the primary spectral hole patterns subject to primary QoS constraints. The second applies real-time scheduling to both primary traffic and virtual secondary calls. The online scheduling algorithms are found to perform well compared to the friendliness upper bound. Extensive simulations of the primary friendly schedulers show the achievable secondary voice capacity for a variety

of parameters compared to non-friendly primary scheduling.

The thesis then considers the variable bit rate (VBR) air interface option for primary transmissions. Offline and online approaches are taken to generate a primary VBR traffic schedule that is friendly to secondary voice calls. The online VBR schedulers are found to perform well compared to the friendliness upper bound. Simulation results are presented that show the effect of the primary traffic load and primary network delay tolerance on the primary network friendliness level towards potential secondary voice traffic.

Finally, secondary user friendliness is considered from an infrastructure deployment point of view. A cooperative framework is proposed, which allows the primary traffic to be relayed by helper nodes using decode-and-forward (DF) relaying. This approach decreases the primary traffic channel utilization, which, in turn, increases the capacity available to potential secondary users. A relay selection optimization problem is first formulated that minimizes the primary channel utilization. A greedy algorithm that assigns relay nodes to primary data flows is introduced and found to perform well compared to the optimum bound. Results are presented that show the primary network friendliness for different levels of primary channel utilization.

# Acknowledgements

After thanking ALLAH the almighty for this big accomplishment, I would like to express my sincere gratitude to my advisor Dr. Terence D. Todd. He not only provided me with continuous support and immense knowledge, but was also very patient and motivating during my Ph.D. study years. His guidance helped me with my research and the writing of this thesis. His insight and the critical thought provoking discussions we had were a huge influence on my way of thinking and his attention to detail showed me how quality work is crafted. Besides my advisor, I would like to thank the rest of my supervisory committee and the members of the examining committee for reading my thesis and for their valuable suggestions and comments. I thank my fellow members of the Wireless Networking Group for their support and comments. I would like to thank my mother for her continuous support and sincere prayers throughout these years. I am also very grateful to my father for his help and guidance throughout my school years, may his soul rest in peace. I would like to thank my brother for being unconditionally encouraging in every stage of my life.

Last but not least, I would like to thank my husband and my two gorgeous sons for their patience, understanding and for putting up with late working nights. It would have been impossible without you.

# Abbreviations

ACK Acknowledgement

AF Amplify-and-Forward

BER Bit Error Rate

BS Basestation

CA Collision Avoidance

CAC Call Admission Control

CBR Constant Bit Rate

CCRN Cooperative Cognitive Radio Network

CDF Cumulative Density Function

CR Cognitive Radio

CRN Cognitive Radio Network

CSI Channel Side Information

CSMA Carrier Sense Multiple Access

CTS Clear to Send

DBP Delay-Bandwidth Product

DF Decode-and-Forward

EDF Earliest Deadline First

FCFS First Come First Serve

FDM Frequency Division Multiplexing

FDMA Frequency Division Multiple Access

GORP Greedy Offline Relay Placement

ILP Integer Linear Program

MAC Media Access Control

MILP Mixed Integer Linear Program

QoS Quality-of-Service

RTS Request to Send

SINR Signal to Interference plus Noise Ratio

STDMA Synchronous Time Division Multiple Access

TB Token Bucket

TDMA Time Division Multiple Access

TPSR Two-Path Successive Relaying

VBR Variable Bit Rate

VN Virtual Node

VoIP Voice over Internet Protocol

# Contents

<b>Abstract</b>	<b>iii</b>
<b>Acknowledgements</b>	<b>v</b>
<b>Abbreviations</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Secondary VoIP Capacity in Opportunistic Spectrum Access Networks with Friendly Scheduling . . . . .	2
1.3 Thesis Organization . . . . .	4
<b>2 Background</b>	<b>6</b>
2.1 Introduction . . . . .	6
2.2 Cognitive Radio Networks Fundamentals . . . . .	7
2.3 Cognitive Radio MAC Layer . . . . .	11
2.3.1 Proposed Cognitive Radio MAC Layer Protocols . . . . .	12
2.3.2 Cognitive Radio MAC Protocol Taxonomy . . . . .	15
2.4 Quality of Service in Cognitive Radio MAC Protocols . . . . .	17

2.4.1	Non-Real-Time Secondary Traffic . . . . .	17
2.4.2	Real-Time Secondary Traffic . . . . .	18
2.4.3	Mixed Secondary Traffic . . . . .	25
2.5	Secondary User Friendliness . . . . .	26
2.6	Cooperative Cognitive Radio Networks . . . . .	28
<b>3</b>	<b>Secondary VoIP Capacity in Opportunistic Spectrum Access Networks with Friendly Scheduling (Constant Transmission Rate)</b>	<b>34</b>
3.1	Introduction . . . . .	34
3.2	Related Work . . . . .	36
3.3	System Model . . . . .	37
3.4	Friendliness Upper Bound . . . . .	42
3.4.1	Optimum Offline Scheduling . . . . .	43
3.5	Online Secondary Friendly Scheduling . . . . .	47
3.5.1	Adaptive Token Bucket Spectrum Hole Shaping Scheduler (TB) . . . . .	48
3.5.2	Virtual Node Scheduler (VN) . . . . .	51
3.6	Simulation Model and Results . . . . .	54
3.6.1	Optimum Offline vs. Online Schedulers . . . . .	57
3.6.2	TB Scheduler: The Minimum Token Rate Threshold $r_{min}$ . . . . .	60
3.6.3	VoIP Codec Configuration . . . . .	62
3.6.4	VN Scheduler: Codec Match vs. Mismatch . . . . .	64
3.6.5	Heavy Tailed Primary Traffic with Different Tail Shape Parameter Values . . . . .	66
3.6.6	Secondary Traffic QoS Requirements . . . . .	68
3.7	Conclusions . . . . .	72

<b>4</b>	<b>Secondary VoIP Capacity in Opportunistic Spectrum Access Networks with Friendly Scheduling (Variable Transmission Rate)</b>	<b>75</b>
4.1	Introduction . . . . .	75
4.2	System Model . . . . .	76
4.3	Optimum Offline VBR Scheduling and Friendliness Upper Bound . . . . .	80
4.4	Online Secondary Friendly VBR Scheduling . . . . .	83
4.4.1	Adaptive Token Bucket Spectrum Hole Shaping VBR Scheduler (VBR-TB) . . . . .	84
4.4.2	Virtual Node VBR Scheduler (VBR-VN) . . . . .	87
4.5	Simulation Model and Results . . . . .	91
4.5.1	Optimum Offline vs. Online VBR Schedulers . . . . .	93
4.5.2	Effect of Primary Traffic Load on Secondary VoIP Friendliness . . . . .	95
4.6	Conclusions . . . . .	103
<b>5</b>	<b>Relay Node Placement for Secondary VoIP Friendliness</b>	<b>106</b>
5.1	Introduction . . . . .	106
5.2	Related Work . . . . .	108
5.3	System Model . . . . .	109
5.4	Offline Relay Node Placement . . . . .	115
5.4.1	Optimal Offline Relay Placement . . . . .	115
5.4.2	Greedy Offline Relay Placement (GORP) Algorithm . . . . .	118
5.5	Simulation Model and Results . . . . .	121
5.5.1	Lower Bound vs. GORP Algorithm . . . . .	122
5.5.2	Online Friendly VBR Schedulers with Relay Support . . . . .	123
5.6	Conclusions . . . . .	135

<b>6</b>	<b>Conclusions and Future Work</b>	<b>137</b>
<b>A</b>	<b>Secondary VoIP Friendliness Using an STDMA Message Delay Model</b>	<b>143</b>
A.1	Secondary VoIP Capacity Estimation using STDMA Delay Analysis . . . .	145
	<b>References</b>	<b>150</b>

# List of Figures

2.1	Spectrum Utilization (© [2006] IEEE [1]). . . . .	8
2.2	Cognitive Radio Cycle (© [2009] IEEE [2]). . . . .	9
2.3	Spectrum Sharing (© [2008] IEEE [3]). . . . .	10
2.4	Classification of CR MAC Protocols Based on Secondary Network Architecture and Spectrum Access Mechanism. . . . .	16
2.5	Time Slot Structure (© [2010] IEEE [4]). . . . .	22
2.6	Secondary Spectrum Access through Cooperation-Based Spectrum Leasing (© [2008] IEEE [5]). . . . .	31
2.7	Spectrum Leasing Schemes (© [2013] IEEE [6]). . . . .	33
3.1	Friendly Basestation Scheduling Example. Primary traffic arrives to the basestation and is placed in the downlink queue (DLQ). The friendly scheduler (FS) transmits these packets subject to deadline constraints. Secondary users are free to monitor the downlink channel and transmit into unused time slots. Friendliness is measured by the number of secondary VoIP connections that can be supported for a given primary traffic scenario. . . . .	39

3.2	Offline Minimum Cost Flow Graph Scheduler. Each edge $(i, j)$ is labeled with an ordered pair $(u_{i,j}, c_{i,j})$ representing the capacity and cost of this edge, respectively. The input and output links have flow capacity of $(K_p + K_s)$ with zero cost. . . . .	46
3.3	Token Bucket Spectral Hole Shaping. To transmit, a packet must obtain a token from the token bucket. A new token is generated every $1/r$ seconds, where $r$ is the set input rate, as long as the number of tokens does not exceed a given threshold. . . . .	49
3.4	Virtual Node Online Friendly Scheduler. Primary traffic arrives to the basestation and is placed in the downlink queue (DLQ). The scheduler schedules real primary traffic and $F_s$ virtual secondary VoIP calls, where contention is resolved in favour of primary traffic. Earliest Deadline First (EDF) is used for real-time traffic scheduling. Real Secondary VoIP calls accessing the channel will make best use of virtual secondary transmission slots (spectral holes). . . . .	52
3.5	Secondary VoIP Capacity vs. Primary Delay Tolerance for OFFLINE–OPT, VN and TB Schedulers. The VN scheduler is in the perfect codec match case. G711-20 ms is used as the real codec configuration. . . . .	58
3.6	Secondary VoIP Capacity vs. Primary Delay Tolerance for OFFLINE–OPT, VN and TB Schedulers. The VN scheduler is in the perfect codec match case. G711-30 ms is used as the real codec configuration. . . . .	59
3.7	Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different values are used for $r_{min}$ . G711-20 ms is used as the codec configuration. . . . .	60

3.8	Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different values are used for $r_{min}$ . G711-30 ms is used as the codec configuration. . . . .	61
3.9	Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different VoIP codec configurations are considered. . . . .	63
3.10	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VN Scheduler. Different VoIP codec configurations are considered (perfect codec match case). . . . .	64
3.11	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VN Scheduler. Different VoIP codec configurations are considered (codec mismatch case (G711-20 ms as the virtual codec configuration)). . . . .	65
3.12	Secondary VoIP capacity vs. Primary Delay Tolerance for the VN Scheduler. Different virtual VoIP codec configurations are considered (codec mismatch case (G711-30 ms as the real codec configuration)). . . . .	67
3.13	Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different values of the Pareto tail shape parameter are considered (G711-30 ms as the codec configuration). . . . .	69
3.14	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VN Scheduler. Different values of the Pareto tail shape parameter are considered (perfect codec match case (G711-30 ms as the codec configuration)). . . . .	70
3.15	Secondary VoIP capacity vs. Primary Delay Tolerance for the VN Scheduler. Different values of the Pareto tail shape parameter are considered (codec mismatch case (G711-20 ms as the virtual codec configuration and G711-30 ms as the real codec configuration)). . . . .	71

3.16	Secondary VoIP capacity vs. Secondary VoIP Call Jitter Requirement for the TB and VN Schedulers (perfect VN match case). G711 is the VoIP codec used with 20 ms and 30 ms packetization periods. . . . .	72
4.1	Secondary VoIP Capacity vs. Primary Delay Tolerance for OFFLINE-VBR-OPT, VBR-VN and VBR-TB Schedulers. The VBR-VN scheduler is in both perfect codec match and codec mismatch cases. G711-20 ms is used as the real codec configuration. G711-30 ms is used as the virtual codec configuration for the VBR-VN scheduler in the codec mismatch case.	96
4.2	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values are used for the mean packet arrival rate ( $\lambda_p$ ) of primary data flows. G711-20 ms is used as the codec configuration. . . .	97
4.3	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the mean packet arrival rate ( $\lambda_p$ ) of primary data flows. Perfect codec match case (G711-20 ms as the codec configuration) . . . . .	98
4.4	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the mean packet arrival rate ( $\lambda_p$ ) of primary data flows. Codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration). . . . .	99
4.5	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values are used for the message length ( $M_p$ ) of primary data flows. G711-20 ms is used as the codec configuration. . . . .	100

4.6	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the message length ( $M_p$ ) of primary data flows. Perfect codec match case (G711-20 ms as the codec configuration). . . . .	101
4.7	Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the message length ( $M_p$ ) of primary data flows. Codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration). . . . .	102
5.1	First Cooperation Phase Scenario . . . . .	111
5.2	Second Cooperation Phase Scenario . . . . .	112
5.3	Normalized Primary Channel Utilization vs. Maximum Number of Open Relays for OFFLINE-RP-OPT Lower Bound and the GORP Algorithm. . .	123
5.4	Low Primary Utilization Case: Primary User and Candidate Relay Layouts within Primary BS Coverage Range. . . . .	125
5.5	Low Primary Utilization Case: Normalized Primary Channel Utilization vs. Mean Normalized Primary Channel Utilization for the GORP Algorithm. Different values of maximum number of open relays ( $R_{max}$ ) are considered. . . . .	125
5.6	Low Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (G711-20 ms as the codec configuration). . . . .	126

5.7	Low Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (perfect codec match case (G711-20 ms as the codec configuration)). . . . .	127
5.8	Low Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration)). . . . .	127
5.9	Average Primary Utilization Case: Primary User and Candidate Relay Layouts within Primary BS Coverage Range. . . . .	129
5.10	Average Primary Utilization Case: Normalized Primary Channel Utilization vs. Mean Normalized Primary Channel Utilization for the GORP Algorithm. Different values of maximum number of open relays ( $R_{max}$ ) are considered. . . . .	129
5.11	Average Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (G711-20 ms as the codec configuration). . . . .	130
5.12	Average Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (perfect codec match case (G711-20 ms as the codec configuration)). . . . .	130

5.13	Average Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration)). . . . .	131
5.14	High Primary Utilization Case: Primary User and Candidate Relay Layouts within Primary BS Coverage Range. . . . .	132
5.15	High Primary Utilization Case: Normalized Primary Channel Utilization vs. Mean Normalized Primary Channel Utilization for the GORP Algorithm. Different values of maximum number of open relays ( $R_{max}$ ) are considered. . . . .	132
5.16	High Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (G711-20 ms as the codec configuration). . . . .	133
5.17	High Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (perfect codec match case (G711-20 ms as the codec configuration)). . . . .	134
5.18	High Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration)). . . . .	134

A.1 Components of STDMA Message Delay © [1990] IEEE [7] . . . . . 145

A.2 Secondary VoIP Capacity vs. Primary Delay Tolerance. The STDMA analysis is compared to the TB and VN schedulers (G711-20 ms codec configuration). . . . . 148

A.3 Secondary VoIP Capacity vs. Primary Delay Tolerance. The STDMA analysis is compared to the TB and VN schedulers (G711-30 ms codec configuration). . . . . 149

# Chapter 1

## Introduction

### 1.1 Overview

Cognitive radio networks (CRNs) define a variety of ways that unlicensed (secondary) users can share underutilized radio bandwidth. This is normally done in a way that is designed to ensure that the licensed (primary) users are unaffected and can result in a more efficient use of the radio spectrum. CRNs have initially focussed on services with low quality-of-service (QoS) requirements, such as web browsing and email. More recently, however, researchers have considered the support of real-time secondary user services such as voice and video. This is considered a challenging task due to the fluctuations in the spectrum availability triggered by primary user activity, but has motivated many researchers to study how CRNs can incorporate real-time QoS support. This area is the topic of the work in this thesis, with a focus on secondary “friendliness” in the context of primary network scheduling.

## **1.2 Secondary VoIP Capacity in Opportunistic Spectrum Access Networks with Friendly Scheduling**

In conventional CRNs, the primary network usually remains unchanged. In some cases however, the primary network operator may wish to accommodate secondary user access to their radio spectrum. In this thesis, we assess the secondary user packet voice (VoIP) capacity when primary basestation scheduling is designed to be secondary network friendly. Friendliness is measured by the number of voice connections that can be supported subject to typical QoS constraints in the presence of delay tolerant primary traffic.

Chapter 3 of the thesis assumes a constant bit rate (CBR) air interface. An offline scheduler is first derived that maximizes friendliness over finite time intervals when the secondary users are transmitting VoIP traffic. The offline scheduler is then obtained using a minimum cost flow graph construction in time complexity that is polynomial in the number of time slots. Two online scheduling algorithms are then compared that achieve various levels of friendliness. The first algorithm operates by having the primary network temporarily shape its residual capacity subject to satisfying its own packet deadline constraints. The second algorithm assumes virtual secondary VoIP calls and applies scheduling to both primary traffic and virtual secondary traffic. The online scheduling algorithms are found to perform well compared to the friendliness upper bound. Results are presented for a variety of parameters that show the degree to which friendly scheduling can improve secondary user VoIP capacity compared to non-friendly primary scheduling. Results show that the more delay tolerant the primary traffic is, the more secondary friendly the primary network can be.

In Chapter 4 of the thesis, the variable bit rate (VBR) air interface option is considered for primary transmissions. As in Chapter 3, offline and online approaches are taken to generate a primary VBR traffic schedule that is friendly to secondary VoIP calls. An offline VBR scheduler is first formulated that maximizes friendliness over finite time intervals. Two online friendly VBR schedulers are then introduced, based on the two online algorithms proposed in Chapter 3. The online VBR schedulers are found to perform well compared to the friendliness upper bound. Simulation results are presented that show the effect of the primary traffic load and primary network delay tolerance on the friendliness level of the primary network towards potential secondary VoIP traffic. Results show that as the primary network traffic load increases, the capacity available for potential secondary VoIP traffic decreases, which, in turn, degrades the secondary friendliness level of the primary network.

In Chapter 5 of the thesis, secondary user friendliness is considered from an infrastructure deployment point of view. A cooperative framework is assumed, where the primary traffic may be relayed by helper nodes. This approach uses a concept that is similar to cooperative CRNs (CCRN), which has been recently proposed to improve primary network transmission. The proposed cooperative framework uses the decode-and-forward (DF) relaying technique to decrease the primary user channel utilization. This, in turn, increases the capacity available to potential secondary users. An optimization problem is first formulated to minimize the primary network channel utilization and provide an optimal solution for the relay node selection problem. A greedy algorithm is then introduced, which is based on mappings of primary data flows to relay nodes. The algorithm is found to perform well compared to the primary channel utilization lower bound. Results are presented that show the primary network friendliness in terms of secondary VoIP capacity, for different levels of

achievable primary channel utilization. The results show that the effect of the proposed cooperative framework on the secondary VoIP capacity substantially depends on the primary user layout and the number of open relays.

### **1.3 Thesis Organization**

Chapter 2 introduces background information related to the focus of this thesis. CRN fundamentals are discussed, which includes the cognitive radio cycle modules and different CRN architectures. The chapter examines the specific characteristics that define the CRN medium access layer (MAC) layer and research efforts are reviewed and classified. A comprehensive review of the research efforts related to QoS-aware CRN MAC layer design, with emphasis on VoIP capacity analysis and call admission control (CAC) techniques is presented. Following this, the limited work done on the concept of secondary user friendliness is reviewed. Finally, a brief overview of the work done in CCRNs is presented.

In Chapter 3, the primary traffic scheduling problem with secondary user friendliness is presented for the CBR air interface case. The chapter first formulates an integer linear program (ILP), whose solution is an optimum offline primary schedule for friendliness towards potential secondary VoIP users. The ILP is then efficiently solved using a minimum cost flow graph construction. Online algorithms are then introduced for secondary user friendliness. The first scheduling algorithm, Token Bucket (TB), uses an adaptive token bucket shaping scheme to create friendly spectral hole patterns for secondary VoIP traffic. The second algorithm, Virtual Node (VN), is based on QoS-aware scheduling of real primary traffic and virtual secondary VoIP traffic. A wide variety of results is presented that include the effects of different secondary user vocoding capabilities on the secondary user capacity. The performance of the system is also investigated under varying primary delay tolerance,

primary arrival process and secondary traffic QoS constraints. An STDMA message delay model is developed in Appendix A, which approximates secondary user friendliness and is used to help validate the simulation model results.

Chapter 4 focuses on secondary VoIP friendliness and primary traffic schedule generation in the case of primary transmissions with a VBR air interface, rather than the CBR air interface case considered in Chapter 3. The chapter first formulates an ILP optimum of-line primary schedule for multiple primary flows to maximize friendliness. Online friendly VBR schedulers are then presented, VBR-TB and VBR-VN, based on the online friendly scheduling algorithms proposed in Chapter 3. The secondary VoIP capacity achieved by the online VBR schedulers is investigated under varying primary traffic load and primary network delay tolerance.

Chapter 5 proposes a cooperative framework that utilizes a DF relaying scheme to reduce the primary channel utilization and increase the capacity available for opportunistic spectrum sharing by secondary users. The chapter presents an ILP formulation of the offline relay selection problem, whose solution is a lower bound on the achievable primary channel utilization. The ILP formulation of the problem is followed by an offline solution given by the proposed Greedy Offline Relay Placement (GORP) algorithm. The performance of the offline relay placement solutions of the optimum bound and the GORP algorithm are studied and compared under varying numbers of open relay nodes. Simulation results present the secondary VoIP capacity achieved by the TB-VBR and VN-VBR schedulers, under the proposed cooperative framework with diversified representative levels of primary channel utilization.

The thesis is then concluded in Chapter 6 with suggestions for possible future work.

# Chapter 2

## Background

### 2.1 Introduction

With the rapid deployment of new wireless devices and applications, the last decade has witnessed a growing demand for wireless radio spectrum. However, it is also known that due to the traditional fixed spectrum assignment policies, the spectral utilization of many licensed networks is very low [8]. This has motivated the use of flexible spectrum management techniques such as *cognitive radio (CR)*, so that scarce channel resources can be more efficiently used. In a cognitive radio network (CRN), wireless devices (known as *secondary users*) are given the capability to adapt their operating parameters and protocols in accordance with interactions they have with the surrounding radio environment. This is done in ways that best protect existing licensed spectrum users (known as *primary users*).

CRNs have been established as a viable networking technology. At the same time, the demand for wireless multimedia services like voice and video streaming has increased significantly. For this reason, supporting real-time quality of service (QoS) is becoming an important requirement. This is a challenging task due to the fluctuations in spectrum

availability triggered by primary user activity. As a result, QoS provisioning in CRNs has been receiving increasing attention in the recent literature.

This chapter starts by discussing the fundamentals of CRNs including the unique characteristics and requirements of the CR media access control (MAC) layer that restrict the direct use of solutions that have been developed for conventional wireless networks. The chapter then presents an overview of some research efforts related to the cognitive MAC layer design for best effort services. CR MAC protocols are then classified according to their CRN architecture and spectrum access procedures. We then review some of the studies that have considered QoS provisioning in CRNs for both non-real-time and real-time services, with an emphasis on supporting voice over CRNs. Previous work on the concept of secondary user friendliness is then discussed. Finally, we present a brief overview of the work done in the new research direction of cooperative CRNs (CCRN).

## 2.2 Cognitive Radio Networks Fundamentals

Conventional fixed spectrum assignment controls interference between different networks and simplifies the hardware design for radio transceivers. Although this is undoubtedly true, the Federal Communications Commission (FCC) [8] reported that licensed spectrum utilization ranges from 15% to 85%, as shown in Figure 2.1, and the traditional regulations prevent access to bandwidth that is temporarily unused. CRNs offer ways to access this unused spectrum in a way that is respectful of its licensed owners.

The term cognitive radio (CR) can be formally defined as follows [8]: “*It is a radio that can change its transmitter parameters based on the interaction with the environment in which it operates*”. A CR node must therefore have two main capabilities, as discussed in the highly cited paper [9]: environment aware capability and re-configurability. The first

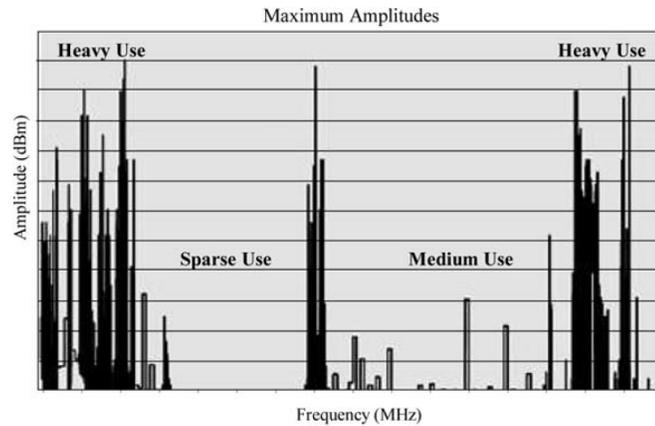


Figure 2.1: Spectrum Utilization (© [2006] IEEE [1]).

capability means that a CR node can capture information from its radio environment. The second means that the CR node can dynamically change certain operational parameters in real time, such as transmission power, modulation scheme and carrier frequency. This is done with two primary objectives in mind: highly reliable communications and efficient utilization of the radio spectrum [9].

In a CRN, two types of networks are involved: the *primary network* and the *secondary network*. The *primary network* is an existing network whose users have legal access to a licensed spectrum band, e.g., cellular networks and TV broadcast networks. The *secondary network*, also known as the cognitive network, accesses the licensed spectrum and opportunistically shares it with the primary users, in ways that best protect the existing licensed spectrum users.

As discussed in [2] [9], a typical cognitive duty cycle (Figure 2.2) includes selecting the best frequency bands, coordinating spectrum access with other users and vacating a

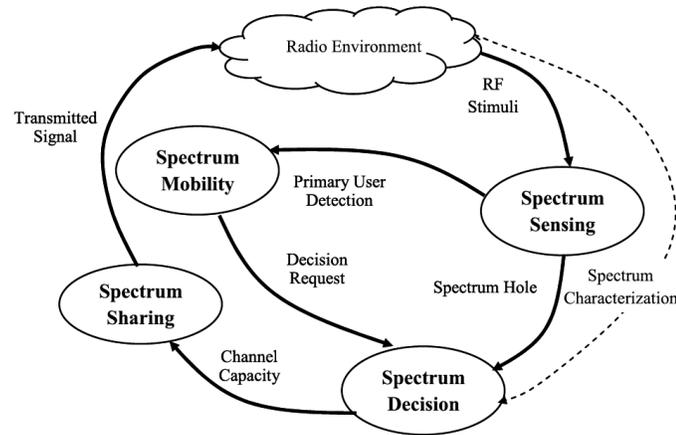


Figure 2.2: Cognitive Radio Cycle (© [2009] IEEE [2]).

frequency band when a primary user appears. This cognitive cycle is supported by the following spectrum management functionalities: *spectrum sensing*, *spectrum analysis*, *spectrum mobility* and *spectrum sharing*. *Spectrum sensing* enables the CR node to monitor the available spectrum bands, capture relevant information and then detect the used/underused frequency bands. *Spectrum analysis* estimates the characteristics of the spectrum bands that were detected through spectrum sensing. *Spectrum mobility* (spectrum handoff) is triggered when the channel conditions become worse or a primary user claims back its spectrum, which results in a change in the secondary user frequency band. *Spectrum sharing* is classified into *intra-spectrum sharing* and *inter-spectrum sharing*. As illustrated in Figure 2.3, *intra-spectrum sharing* takes scheduling decisions that map secondary transmissions to available spectrum bands based on spectral characteristics and the secondary user requirements. *Inter-spectrum sharing* coordinates spectrum sharing between multiple secondary networks.

CRNs offer different paradigms for spectrum sharing between the primary and secondary networks. Based on the type of the shared spectrum bands, spectrum sharing can be classified into three main types: *spectrum underlay*, *spectrum overlay* and *spectrum*

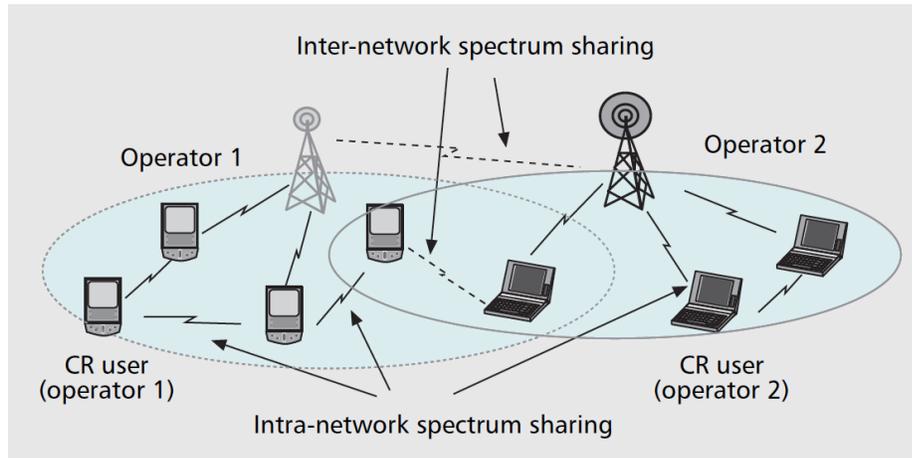


Figure 2.3: Spectrum Sharing (© [2008] IEEE [3]).

*interweave* [10]. In a *spectrum underlay*, secondary users access the spectral band simultaneously with the primary users, but subject to interference thresholds that are sufficient to prevent primary user disruption. In a *spectrum overlay*, secondary nodes can simultaneously share the spectrum with primary users, but with no preset interference threshold. Instead, to compensate for the interference caused by the secondary transmissions, secondary nodes use their knowledge of the primary traffic to either actively assist in relaying primary messages or to cancel out interference effects. In a *spectrum interweave*, secondary nodes may identify and exploit spatial and temporal spectrum white space, referred to as *spectrum holes*, in a non-intrusive manner [11]. An example of temporal interweave spectrum sharing is when free primary user time slots are dynamically detected and used by the secondary users. This is the type of spectrum sharing considered in this thesis.

As in conventional wireless networks, a given primary or secondary network can be infrastructure-based or infrastructure-less. The difference between the two is that the former has a central base station while the latter is typically an ad-hoc network with no central control. In this thesis, we consider collocated infrastructure-based primary and secondary

networks.

On the other hand, the evolving fifth generation (5G) cellular wireless networks offer solutions to the increasing demand for high data rates and mobility required by new wireless applications. To overcome the challenges faced by the existing cellular networks, wireless system designers have started research on 5G wireless systems that are expected to be deployed beyond 2020 [12] [13]. Meanwhile, CR is considered one of the key promising technologies that can be deployed to deliver the 5G requirements. Therefore, several research efforts, e.g., [14] and [15], have recently proposed the idea of designing a 5G CRN that uses the CR concept to improve the spectrum utilization of the existing radio spectrum available for 5G wireless communication systems. The challenges facing the commercial deployment of CRNs is discussed in detail in [16], which investigates the real world applicability of a large volume of existing research in the field of CRNs.

## **2.3 Cognitive Radio MAC Layer**

Cognitive radio procedures enable the secondary users to share licensed spectrum in space and time with no or little interference to primary users. Based on the CR protocol stack presented in [17] [18], the PHY layer is responsible for spectrum sensing and physical tuning of frequency operating parameters. The MAC layer is responsible for spectrum analysis, spectrum sharing, spectrum mobility as well as scheduling the spectrum sensing intervals. Hence, the design of CR MAC protocols imposes new challenges that are not considered in conventional MAC protocols, which use fixed parameters suited for static spectrum policies. In this section some of the proposed CR MAC protocols for best effort services are reviewed and classified according to their architecture and spectrum access scheme.

### 2.3.1 Proposed Cognitive Radio MAC Layer Protocols

Recently, several MAC protocols have been proposed for secondary nodes, offering techniques for solving the intra-spectrum sharing problem as well as ways for managing the spectrum sensing timing. This section provides a quick review of some of the recent proposals in this area, based on the literature surveys provided in [17] [19].

CR MAC protocols based on a carrier sense multiple access (CSMA) channel model were considered in [20–26]. Reference [20] proposed a simple extension of the IEEE 802.11 CSMA/CA (carrier sense multiple access with collision avoidance) protocol using distributed channel assignment (DCA). This protocol uses multiple transceivers with a dedicated channel for signaling. It also utilizes spectrum pooling to enhance spectral efficiency by reliable primary network activity detection. Reference [21] developed a CSMA/CA based distributed open spectrum sharing (DOSS) MAC protocol, which consists of five main functionalities: primary user detection, setting operational frequencies for control and data use, spectrum mapping, spectrum negotiation and data transmission. In reference [22], a CSMA/CA based MAC protocol was proposed where every secondary node has its own home channel (Hch). A given secondary node can change its Hch if it is noisy, claimed back by a primary user or if it conflicts with another secondary user Hch. A distributed CSMA/CA based MAC protocol [23] exploits statistics of spectrum usage for channel access decision making. Before each transmission, a negotiation of transmission parameters between a sender and a receiver is performed through a control channel. A centralized CSMA-based MAC protocol [24] considered underlay spectrum sharing and interference control techniques. RTS/CTS handshaking is used for both primary and secondary transmissions. Reference [25] introduced the notion of a secondary user group (SUG), where the secondary nodes are divided into clusters and a delegate node is set for every cluster.

This node is responsible for exchanging information with other clusters and to contend for spectrum bands for its own cluster. Only one user in a given cluster can transmit in a given time slot as a way of contention resolution. A hardware-constrained (HC) multi-channel CR MAC scheme [26] conducts efficient spectrum sensing and spectrum access decisions in ad-hoc CRNs. This protocol considered the case when secondary users have limited spectrum sensing and transmission capabilities.

Synchronized time-slotted CR MAC protocols were studied in [27] [28]. Reference [27] proposed a cognitive MAC protocol (C-MAC), which aimed at higher aggregate link throughput and robustness to spectrum change using multiple transceivers. C-MAC includes two key concepts: the rendezvous channel (RC) and the backup channel (BC). The RC is selected as the channel that can be used for the longest time without interruption among all available choices. It is used for node coordination, primary user detection as well as multi-channel resource reservation. The BC, determined by out-of-band measurements, is used to immediately provide a choice of alternate spectrum bands when a primary user appears. A time-slotted synchronized MAC (SYN-MAC) was also proposed in [28] that uses a dedicated radio for control messages. A second transceiver is used for data traffic. At the beginning of each time slot, the CR users tune their dedicated control radios to the specified channel, and the users that wish to initiate a data transfer send out a beacon. Interested neighbours respond with their own list of available channels, and further communication is carried out in one of the available channels.

Spectrum sensing scheduling was studied in [29–32]. The optimal trade-off in sensing time setting was investigated for the single channel and multi-channel cases in [29] and [32], respectively. The channel sensing order problem was also investigated for the single-user case in [30] and the two-user case in [31].

As for standardization efforts, IEEE 802.22 [33], is a centralized standard that uses base stations for spectrum access and sharing. The base station manages its own cell and all associated CR users. In the downstream (DS) direction, the IEEE 802.22 MAC uses time division multiplexing, while in the upstream (US) direction, demand assigned TDMA is used. The standard specifies schemes for spectrum sensing, spectrum recovery as well as coexistence with users.

### **Cognitive Radio MAC Design for Best Effort Services**

Best effort secondary traffic has no QoS requirements; hence no QoS guarantees are provided by the secondary MAC scheduler. However, some work has been aimed at maximizing secondary network throughput subject to primary network constraints. For example, in [34] an opportunistic scheduling policy was proposed to maximize the throughput utility which reflects the QoS of the secondary users subject to maximum collision constraints with the primary users. In [35], a scheduling optimization problem was formulated to maximize the long-term average throughput of an ad-hoc secondary network while maintaining fairness among the secondary nodes, subject to primary interference constraints and a secondary link signal to interference plus noise ratio (SINR) threshold. However, no minimum rate requirements were considered for the secondary links. In reference [36], heuristic control policies were proposed with the objective of carrying out optimal sensing and maximizing the expected reward of secondary users subject to an energy constraint. The secondary user reward was defined as a function of delay, energy costs and throughput gain. Reference [37] also developed an optimal channel access strategy based on a partially observable Markov decision process (POMDP), which optimizes the performance of secondary users while limiting the interference perceived by primary users.

Other proposals provide QoS-aware analyses for a specific secondary network utility (e.g., throughput, delay, etc.), but with no QoS guarantees. For example, [38] and [39] considered a throughput analysis for CRNs to improve the reliability and accuracy of the spectrum sensing techniques and to maximize the spectrum utilization. Reference [38] also presented a packet transmission delay analysis, which provided insightful guidelines to enable QoS support over CRNs for subsequent research. This study considered a time-slotted multi-channel system model, where every secondary node is equipped with two transceivers for control information and data transmission purposes. Secondary nodes use a dedicated unlicensed control channel to exchange information about spectrum sensing (co-operative spectrum sensing) and to perform handshaking before data transmission, which is done in two phases (reporting and negotiation).

### **2.3.2 Cognitive Radio MAC Protocol Taxonomy**

Based on the literature survey provided in [17], CR MAC protocols can be categorized according to the secondary network architecture or the spectrum access mechanism. The secondary network architecture can be *centralized* or *distributed*. In a *centralized* CRN, a basestation coordinates the spectrum access procedures. In a *distributed* CRN, such as an ad-hoc network, the spectrum access procedure is shared by all users. For this reason, distributed CRNs have gained more interest, although spectrum management in such networks is more complicated.

Another way to classify CR MAC protocols is according to the employed access mechanism; for instance, whether the CR MAC protocol is *random access*, *time-slotted*, or a *hybrid* MAC protocol. *Random access* MAC protocols are generally based on the carrier

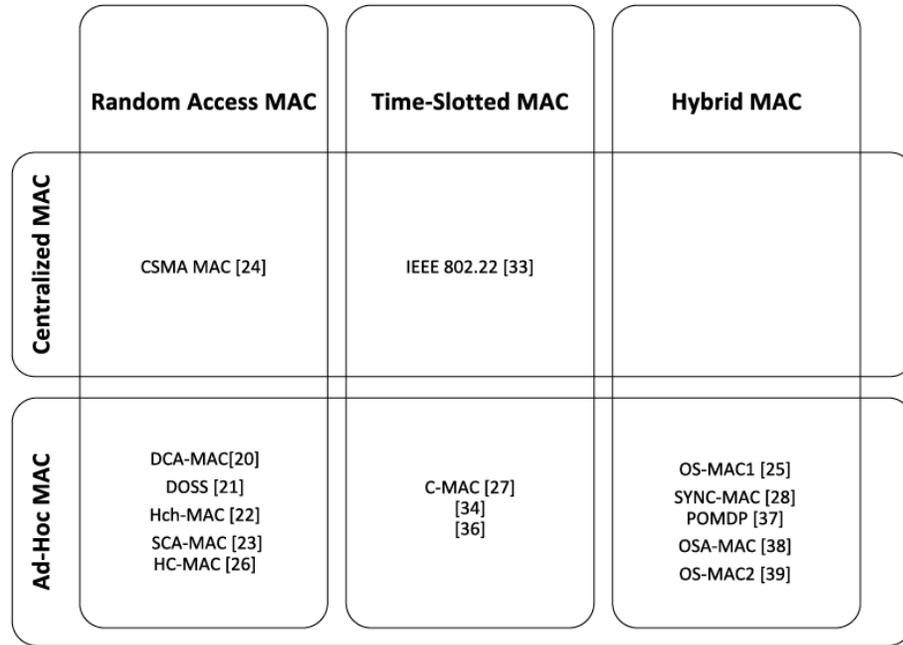


Figure 2.4: Classification of CR MAC Protocols Based on Secondary Network Architecture and Spectrum Access Mechanism.

sense multiple access with collision avoidance (CSMA/CA) principle. The CR user monitors the spectrum band to detect when there is no transmission from the other cognitive users and transmits after a back-off duration to prevent simultaneous transmissions. *Time-slotted* protocols need network-wide synchronization, where time is divided into slots for both the control channel and data transmission. *Hybrid* protocols use a partially slotted transmission, in which the control signaling generally occurs over synchronized time slots, while data transmission may use random channel access. In a different approach, the durations for control and data transfer may have predefined durations. Within each control or data period, random channel access is used. Figure 2.4 categorizes the previously reviewed cognitive MAC protocols based on both the secondary network architecture and the access mechanism.

## 2.4 Quality of Service in Cognitive Radio MAC Protocols

A CR MAC protocol can support QoS if the scheduling algorithm in the intra-spectrum sharing module is designed to be QoS-aware. QoS can have different requirements depending on the application, such as delay, jitter, packet loss and minimum bit rate constraints. As discussed in [18], supporting QoS in secondary networks is more challenging than in conventional wireless networks, because the secondary network services may always be interrupted when the primary users reclaim the spectrum. The secondary network performance may therefore degrade severely if this happens too frequently. In this section we review some of the proposed QoS-aware scheduling techniques for CR MAC protocols. We categorize this research efforts according to the type of secondary traffic (non-real-time or real-time) with a focus on supporting voice services over CRNs, voice capacity analysis and CAC procedures.

### 2.4.1 Non-Real-Time Secondary Traffic

Non-real-time secondary traffic usually has throughput or minimum bit rate requirements for the secondary traffic flows. To guarantee a certain QoS level for loss sensitive applications, the secondary network has to be equipped with a QoS-aware resource allocation scheme that avoids channel saturation. Reference [40] proposed a cognitive MAC protocol in which secondary users are divided into several nonoverlapping groups. An optimization problem was formulated to maximize the bandwidth utilization of the shared spectrum subject to minimum bit rate QoS requirements of the secondary flows and with fairness maintained. In [41], a distributed multi-channel power allocation problem was formulated as a non-cooperative game, which minimizes the secondary network energy consumption, subject to a target rate for every secondary user. A game theoretic dynamic spectrum access

(DSA) was proposed in [42], which includes a throughput and delay analysis for non-real-time secondary traffic. The data transfer occurs in predetermined time slots, while the control signaling uses a random access scheme, making it a hybrid protocol. This MAC is cluster based and the game policy in each cluster is managed by a central entity within the cluster. The proposed MAC protocol has high spectrum utilization, collision free spectrum access with minimum bit rate and fairness guarantees for secondary users. Secondary spectrum access with secondary SINR and primary interference constraints was studied in [43] for secondary users with different non-real-time traffic priority classes. A nonlinear optimization problem was formulated with the objective of maximizing the total transmission rate of all secondary users, subject to secondary user QoS requirements. When not all the secondary links can be supported with their QoS requirements, the spectrum access opportunity is designed to be proportional to the user's priority traffic class.

Recently, [44] presented one of the first studies of the performance of request-response type services such as web browsing over CRNs. A response time analysis of elastic data traffic was introduced under three service disciplines (shortest processor time without preemption, shortest processor time with preemption and shortest remaining processing time) for a single channel centralized CRN. One of the findings was that the mean duration of the transmission interruptions has an impact on the mean response time even when the long-term channel availability and the system load remain unchanged.

### **2.4.2 Real-Time Secondary Traffic**

Real-time secondary traffic requires both minimum bit rate and delay constraints. To guarantee an acceptable level of QoS for applications such as voice and video streaming, deploying a CAC technique in the intra-spectrum sharing MAC module is required to

avoid network saturation and QoS violations. However, fluctuations in spectral availability caused by the presence of primary users make providing the required QoS a challenging task. This has motivated a lot of research to design and analyze CR MAC protocols that support delay sensitive secondary traffic [45–48]. Since voice is the most common real-time service, several research efforts have focused on voice service support over CRNs. In reference [49] the delay-bandwidth product (DBP) was introduced for variable bandwidth channels. Based on the DBP index, secondary users select the optimal channel that maximizes throughput and guarantees their QoS requirements. The proposed scheme aims to satisfy a specific target rate and maximum tolerable delay requirements for secondary voice traffic flows.

To the best of our knowledge, [50] was the first paper to provide a comprehensive study of VoIP QoS parameters to enhance call quality over CRNs. Under interweave spectrum sharing and imperfect spectrum sensing, a primary network with uniform constant bit rate traffic was assumed to share a single channel with a secondary VoIP network. Modifications to the choice of sensing and transmission intervals were proposed. The proposed framework addressed the concerns of interference in the case of primary user activity while achieving maximum throughput with acceptable call quality for the secondary users. An analysis of the proposed enhancements showed that even with zero tolerance for primary user interference, acceptable VoIP performance was maintained. The study in [50] was extended in [51] to further optimize the transmission interval in the cognitive MAC layer design to enhance VoIP call quality while protecting primary users.

VoIP over CRNs was also studied along with a primary user activity analysis [52] [53]. In both studies, online scheduling algorithms for the cognitive MAC layer were proposed

that adapt the VoIP transmission codec based on the primary user activity patterns. Specifically, [53] studied the case when both the primary and secondary networks transmit VoIP calls under a single channel interweave spectrum sharing paradigm. VoIP traffic was modelled as intervals of talk spurts and silence periods with delay, jitter, packet loss, MOS and R-Factor criteria. The effect of changing the sensing and transmission intervals on the performance of secondary VoIP calls was also studied. This study also proposed a centralized secondary scheduling scheme that adaptively changes the VoIP codec rate based on the primary user activity. Simulation results confirmed the VoIP throughput maximization along with enhanced QoS under the proposed scheduling algorithm. Other efforts that studied voice over CRNs include [54–56].

### **VoIP Capacity Analysis**

Voice capacity analysis is used to obtain the maximum number of secondary voice users that can be supported (admitted into the system) with QoS guarantees. Reference [57] proposed one of the earliest protocols that tackle the issue of QoS support for CRN voice services, which was extended in [4] to further study the impact of the primary user activity. In [4], a TDMA QoS-aware scheduling for real-time traffic that analyzes the voice capacity of secondary users was considered. A single channel was shared between a small coverage primary network and a distributed CRN under interweave spectrum sharing. Synchronized secondary users sense the channel at the beginning of every time slot and exploit the channel availability only if the time slot is idle. As shown in Figure 2.5, a time slot is divided into four parts: sensing, contention, transmission and ACK. Based on this structure, two distributed time-slotted CR MAC control schemes were proposed to support the cognitive

voice service. The first MAC protocol is contention based, where the secondary users contend for idle time slots using the backoff contention mechanism in the contention part. The second MAC protocol is contention-free where the contention part is divided into mini-slots, each assigned to a secondary user. After a time slot is sensed idle (sensing part) and a secondary user is chosen as a sender (contention part), the transmission and ACK parts of the time slot structure are then used for secondary traffic transmission and acknowledgement of reception, respectively. The impact of the primary user activity on the secondary voice capacity was analyzed for both correlated and independent channel state models. In the correlated channel state model, the channel busy/idle states over neighbouring slots are correlated rather than independent. This extended the work proposed in [57], which considered an independent channel state model. Reference [4] also considered constant rate voice traffic for secondary users with delay and packet loss rate requirements. Markov chain models were developed to obtain the secondary voice capacity of the two proposed MAC schemes. The number of secondary users is gradually increased and the simulation is repeated until the packet loss ratio is larger than the packet loss rate target. The maximum number of secondary users which yields a packet loss ratio less than the packet loss rate target is defined as the system capacity. Results showed that as the number of primary users increases, the secondary voice capacity decreases and the capacity of the contention free MAC protocol is larger than that of the contention based MAC protocol. Also, as the correlation of the channel increases, the secondary voice capacity decreases, since the channel tends to remain busy for longer time periods. In this thesis, we use the method described in [4] to determine the secondary VoIP capacity obtained from simulation.

Secondary VoIP capacity was also analyzed using a discrete-time Markov chain where

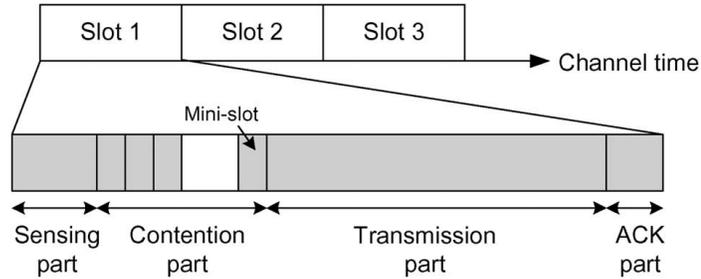


Figure 2.5: Time Slot Structure (© [2010] IEEE [4]).

spectrum sensing errors do not occur [58], which was extended to include energy-detection-based spectrum sensing under the constraint of imperfect spectrum sensing in [59]. To the best of our knowledge, [59] was the first reference to analyze VoIP capacity over CRNs where spectrum sensing errors occur. This study proposed a centralized multi-channel CRN that supports VoIP services for a correlated primary occupancy two-state channel model. VoIP traffic was modelled as a Markov-modulated Poisson process (MMPP) [60], with a packet dropping probability constraint. A new method to find the minimum target-detection and false-alarm probabilities was proposed. A 2-D discrete-time Markov chain queueing model was used to derive the secondary VoIP capacity in the case of imperfect spectrum sensing. The authors experimented with gradually increasing the number of VoIP users and computed the secondary VoIP capacity as the maximum number of admitted users that satisfy the dropping probability threshold. It was concluded that the detection and false-alarm probabilities have to be adjusted based on the packet dropping probability. As the number of secondary users sensing the channel increases, the channel detection probability increases, which, in turn, enhances the sensing accuracy and increases the secondary VoIP capacity of the system. The results showed that the average throughput of the system grows linearly as the number of VoIP users increases and the maximum throughput of the system increases as the sensing accuracy improves. The study in [59] was then extended to

consider VoIP traffic on multiple channels, taking into account VoIP packet retransmissions [61].

Secondary VoIP capacity was also studied along with different CRN channel access protocols in [62] [63] for independent and correlated primary occupancy channel models. Both studies considered VoIP end-to-end delay and the maximum allowable delay violation probability. Reference [62] analyzed VoIP capacity for a multi-channel centralized CRN with an FCFS (first come first served) service discipline at the secondary basestation. Constant bit rate secondary VoIP traffic was modelled as an on-off Markov fluid model [64]. A model based on the effective bandwidth (EB) [65] and the effective capacity (EC) [66] was used to derive the secondary VoIP capacity. This was also obtained from simulation as the maximum number of secondary VoIP calls that permits an acceptable delay violation probability. In the performance evaluation, the effect of varying the average durations of the channel availability was studied. The results showed that the average duration of the channel unavailability to secondary VoIP users has a more significant effect on the secondary VoIP capacity than the average duration of the channel availability. The study in [62] was then extended in [63] to analyze secondary voice capacity for both distributed and centralized multi-channel CRN models. In reference [63] the constant rate voice capacity of a multi-channel CRN was analyzed for both slotted-ALOHA and round-robin channel access coordination schemes. Possible analytical extensions to incorporate sensing errors were also discussed. Secondary VoIP capacity was also studied along with on-demand sensing [67] and cooperative sensing over multiple TV spectrum bands [68].

### **Call Admission Control (CAC)**

Call admission control is responsible for admitting new voice calls based on the available resources. However, CAC strategies developed for conventional wireless networks do not fully address the different requirements of the cognitive MAC layer. This has motivated many researchers to use CRN VoIP capacity analysis to develop CAC strategies. For example, reference [69] proposed a CAC framework for secondary voice traffic QoS provisioning based on the secondary voice capacity analysis presented in [63]. Two CAC procedures were developed for a single channel CRN with slotted ALOHA network coordination.

Other CAC research efforts based on secondary capacity analysis include [70–74]. Unlike the work previously discussed in Section 2.4.2, reference [70] relaxed the assumption of a fixed number of secondary users. This reference was one of the first papers to provide a comprehensive study of the impact of primary resource occupancy information (PROI) for the admission criteria. VoIP traffic was assumed in both the primary and secondary networks. The secondary VoIP traffic was modelled as an on-off source with exponentially distributed call durations and a Poisson call arrival process. The packet dropping probability, mean packet delay and call blocking probability were evaluated for both available and unavailable PROI scenarios. Joint call and packet level teletraffic models as well as CAC strategies were developed based on a system capacity analysis. Numerical results showed that including PROI is an effective means to improve system performance, especially for low to moderate traffic loads. The study in [70] was extended by adopting an ON/OFF primary arrival traffic model under unreliable spectrum sensing [71] [73]. CAC policies based on joint packet and connection level analysis were also designed to increase VoIP Erlang capacity, which is defined as the maximum value of the offered VoIP traffic load for

which all the QoS requirements are met [72].

Reference [74] employed the analytical methodology proposed in [73] and extended the study in [72] to propose a differentiated resource occupancy aware CAC scheme with packet buffering and perfect spectrum sensing. The proposed CAC algorithm allows efficient spectrum usage by taking into account the effective resource occupancy of different users. Based on a joint connection and packet level analytical model, the performance of the proposed differentiated traffic CAC scheme was investigated. The proposal showed significant improvements in the VoIP Erlang capacity and spectrum efficiency compared to conventional benchmark CAC procedures.

### **2.4.3 Mixed Secondary Traffic**

Heterogeneous classes of secondary traffic require a prioritized cognitive MAC layer, which considers different levels of QoS requirements. For example, reference [75] proposed a distributed cognitive MAC protocol that maintains high efficiency and fairness among secondary users with different traffic classes. The proposed MAC layer was evaluated in terms of non-real-time traffic throughput and real-time traffic dropping rate. Reference [76] proposed admission control, rate control and resource allocation modules for point-to-multipoint CRNs, which guarantees the QoS requirements of different kinds of admitted connections (CBR, non-real-time and real-time). In a highly cited paper [77], call level dynamic spectrum access schemes have been proposed for centralized and distributed CRNs. Two priority levels were considered among secondary users and the effect of channel reservation for the higher priority was investigated. Also, in [78], real-time and non-real-time secondary users were considered with the spectrum access priority given to

the real-time users. A MAC channel allocation strategy was proposed and the QoS parameters of each class of users were analyzed. Other efforts that considered QoS provisioning for heterogenous traffic over CRNs include [79–83].

## 2.5 Secondary User Friendliness

A conventional CRN typically assumes that there is no modification to the operation of the primary network [11]. However, in some scenarios the primary network operator may wish to accommodate secondary user access, through informal or formalized relationships. To the best of our knowledge, [84] is the first reference to consider this approach and propose the concept of secondary user friendliness. Reference [84] addresses the option of having the primary network utilize its spectrum in a way that is more compatible with a future secondary network, i.e. having a primary user that utilizes its spectrum in a way that is *friendly* to potential secondary users. A modification to the primary user TDMA scheduling algorithm was proposed to share the spectrum with an ad-hoc secondary network under a seamless interweave spectrum sharing paradigm. The proposed scheduling algorithm allows the primary network to operate in a friendly manner towards the secondary users while still satisfying the primary users' own QoS requirements. This is accomplished by delaying the primary user's transmissions when channel conditions are poor and by sending them either at a later time when channel quality has improved or when the primary user's backlog queue approaches overflow. This improves the overall primary network channel usage, thus increasing spectrum hole availability for the secondary network. A queuing model was developed to analyze the performance of the proposed scheme, in terms of the idle time slot probability, the average packet delay, the head of line delay, the throughput and the packet drop rate. The performance evaluation showed that the secondary network

maximum throughput was increased at the expense of increasing the primary network delay without affecting the primary network throughput. Primary data traffic was considered with tuneable packet delay and drop rate QoS parameters, while secondary traffic was assumed to be best effort, i.e. no QoS requirements were considered for secondary transmission links. Hence, friendliness in [84] was proposed specifically for best effort secondary traffic and is a bi-product of improved primary network bandwidth usage when there is sufficient primary traffic delay tolerance. In Chapters 3 and 4, we take a different approach to secondary user friendliness based on primary basestation scheduling.

Another research effort on secondary user friendliness under interweave spectrum sharing was presented in [85]. This study proposed a resource allocation scheme that uses delayed time periods of primary users to transmit secondary users' data packets subject to primary user QoS requirements. The proposed scheme directly favors secondary traffic at high primary traffic loads to avoid secondary network starvation, unlike [84]. A delay analysis of the primary and secondary packets under the proposed scheme was first derived using an  $M/G/1$  queueing analysis based on that presented in [86]. The proposed spectrum assignment approach was numerically evaluated for various types of applications (data, audio and video) with different QoS requirements. The results showed that it is possible to provide some QoS to secondary users with no absolute guarantees, while preserving the acceptable quality requirements of the primary users. However, the authors claimed that this might not be the case in a realistic implementation of the system.

Reference [87] proposed an underlay spectrum sharing and leasing model for CRNs, which is implicitly friendly to potential secondary users. In this study, the primary user adjusts its willingness to lease the spectrum by tuning the amount of interference it can bear. The secondary users attempt to achieve maximum possible throughput, while not

violating the interference limit. A game theoretic framework was formulated where the primary users control the frequency pricing and the demand for spectrum access based on their QoS requirements. The system's Nash equilibrium was achieved with no significant interaction between the two networks. Performance evaluation was presented in terms of the interference level at the primary users and the secondary network maximum throughput under varying channel quality. Since better performance with less energy consumption was achieved for the primary system and the secondary transmission requirements were satisfied, the proposed spectrum leasing model is a win-win game for both systems.

## 2.6 Cooperative Cognitive Radio Networks

Over the years, two main approaches to CRNs have emerged: the *commons* and *property-rights* models [88]. In the *commons* model, the primary users are oblivious to the presence of secondary users, which sense the radio environment in search of spectrum holes and exploit the detected transmission opportunities. This includes most of the commonly studied *interweave* and *underlay* spectrum sharing CRNs. On the other hand, in the *property-rights* model, the primary users may decide to lease part of their spectrum to secondary users in exchange for appropriate remuneration (e.g., spectrum pricing, cooperative spectrum leasing). This has mainly been studied under the *overlay* spectrum sharing model. In an *overlay* spectrum sharing model, the secondary users use signal processing and coding to maintain or improve the communication of primary users while also obtaining some additional bandwidth for their own communication [10]. This cooperation framework allows higher transmission rates for primary users and higher capacity for secondary users resulting in incentives for both networks to cooperate. Spectrum pricing was also modelled in some studies to increase the primary network's incentive to cooperate. Applying

cooperative communication within a CRN results in a *cooperative cognitive radio network* (CCRN), where secondary users or access points are leveraged to relay primary traffic, which reduces the primary channel utilization and increases the available secondary traffic capacity. The CCRN model relaxes the assumption of oblivious primary users in the original CRN definition [11], which, in turn, allows various forms of cooperation. CCRN research efforts have covered a broad range of topics which include: 1) different relaying techniques (decode-and-forward (DF) [5, 89, 90], amplify-and-forward (AF) [91], two-path successive relaying (TPSR) [92] [6], etc.); 2) different spectrum sharing domains (time domain [5, 6, 89–92] and frequency domain [93] [94]); 3) different cooperative spectrum leasing models based on whether the primary user is only aware of the secondary cooperation [95] [96] or cooperative as well (i.e., engaged in an active cooperative communication with the secondary network [5, 6, 89–94]). In Chapter 5, we propose a DF-based cooperation framework for the primary network, which enhances the primary network transmission rates and resembles the CCRN spectrum leasing model. Hence, in this section we focus on some research efforts that studied the CCRN spectrum leasing model of cooperative primary and secondary users, utilizing the time domain DF relaying technique.

Reference [5] considered a single channel shared between a primary transmission link and an ad-hoc secondary network. In this study, a DF cooperative spectrum leasing solution was proposed and investigated, where secondary nodes earn spectrum access in exchange for cooperation with the primary link. In the proposed solution, secondary users have the option to cooperate with the primary user based on the required amount of cooperation and the corresponding fraction of time leased for secondary transmissions. Under the proposed spectrum leasing model, a given time slot is divided into three subslots, as shown in Figure 2.6. The parameter  $\alpha$  divides the time duration between primary and secondary spectrum

access, and  $\beta$  divides the time duration between relaying primary traffic and secondary data transmissions. In the first subslot, the primary user transmits its data to a selected set of cooperative secondary relay nodes (Figure 2.6-(a)), which, in turn, relay the primary user's data to the primary basestation in the second subslot (Figure 2.6-(b)). Finally, the third subslot is used for active secondary data transmissions (Figure 2.6-(c)). The proposed spectrum leasing framework is based on improving the primary users QoS level via secondary cooperation, instead of frequency pricing. The problem was formulated using Stackelberg games and analyzed for both instantaneous channel side information (CSI) and long-term CSI scenarios. The analysis derived the optimal values for the subslot durations ( $\alpha$  and  $\beta$ ) as well as the conditions under which it is advantageous for the primary user to lease the bandwidth for a fraction of time to the secondary network. The results proved that the proposed model is an effective paradigm for CCRN.

Reference [90] extends the work proposed in [5] to consider multiple primary transmission links. In this study, a cooperative communication-aware spectrum leasing framework was proposed for collocated infrastructure-based primary and multiple secondary networks. The secondary access points were selected as relays instead of the secondary users (as in [5]). In the proposed spectrum leasing model, the primary network uses the secondary access points as cooperative relays to enhance the performance of all networks using a DF relaying technique. A Stackelberg game was formulated where the primary network decides the optimal strategy for the relay selection and spectrum pricing, while the secondary networks determine the length of spectrum access time to purchase from the primary network. At a Nash equilibrium point, all networks maximize their utilities in terms of transmission rate and revenue/payment. The utilities of the primary and secondary networks under the proposed scheme were evaluated under varying primary traffic load and for

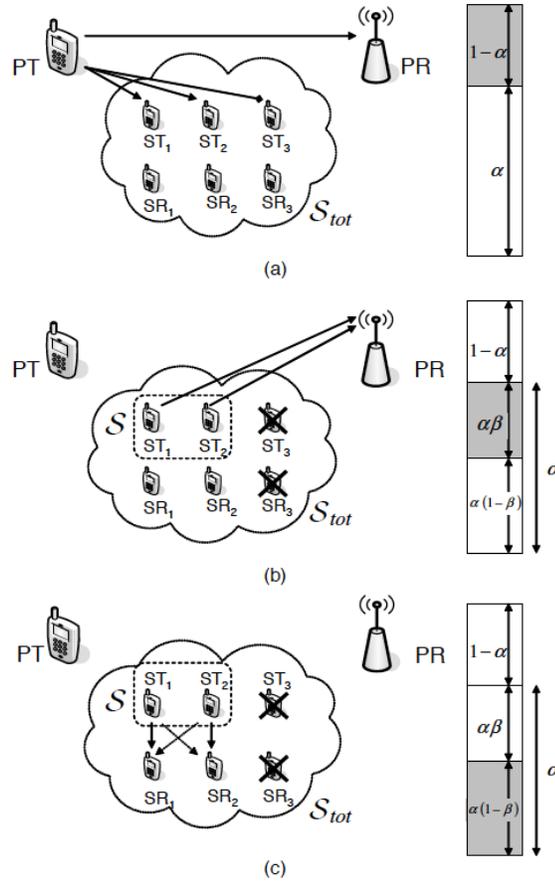


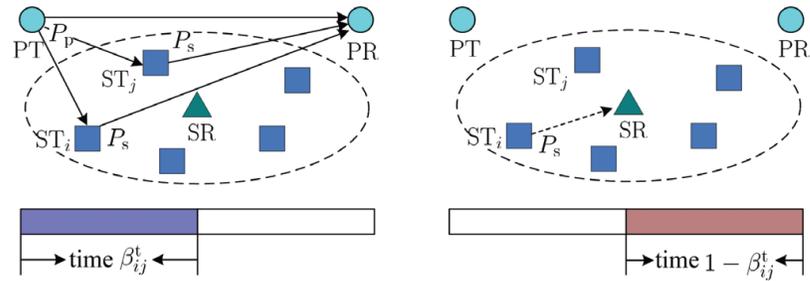
Figure 2.6: Secondary Spectrum Access through Cooperation-Based Spectrum Leasing (© [2008] IEEE [5]).

different numbers of secondary networks. Compared to a conventional non-spectrum leasing scheme and a non-cooperative spectrum leasing scheme, the proposal achieved higher utilities for the primary and secondary networks by exploiting cooperative transmission.

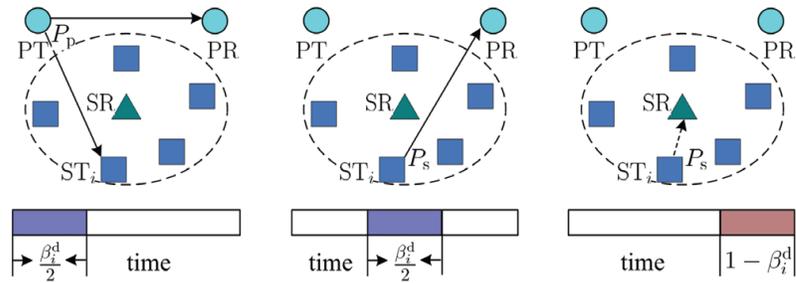
In reference [6], a cooperative spectrum leasing scheme was proposed for multi-user CRNs sharing the spectrum with a single primary link, where secondary users adaptively switch between DF relaying and TPSR schemes. As shown in Figure 2.7,  $\beta^d$  and  $\beta^t$  divide a given time slot between primary and secondary spectrum access in the DF and TPSR schemes, respectively. As shown in Figure 2.7-(b)), the DF relaying mode works similar

to the model proposed in [5], utilizing one secondary relay node  $ST_i$ . In the TPSR mode, two secondary relay nodes,  $ST_i$  and  $ST_j$ , are selected to relay the primary data to the primary basestation simultaneously with the primary data transmission, as shown in Figure 2.7-(a). The primary basestation gets the original signal with superimposed versions of it, which increase the achievable primary transmission rate. Under both relaying schemes, the primary transmission time is shortened and the cooperative secondary user(s) are rewarded by leasing the spectrum for secondary data transmission for the remaining part of the time slot, as shown in Figure 2.7. For a given set of secondary users, the one that can achieve the largest secondary rate while satisfying a primary rate target is scheduled for spectrum leasing. An optimization problem was formulated to obtain the time allocation for primary and secondary spectrum access that maximizes the predefined utilities. Performance results verified the efficiency of the proposed adaptive spectrum leasing scheme compared to the stand-alone DF and TPSR mechanisms.

Besides improving the primary user transmission rates, CCRN spectrum leasing models have been also proposed to maintain the primary user transmission. For example, [89] proposed a DF-based cooperative spectrum sharing scheme, where the primary data is cooperatively transmitted to compensate for the interference caused by secondary data transmissions. Under the proposed scheme, the primary system is only aware of a “DF relaying with relay selection” operation mode. However, it is ignorant of whether the relays are primary or secondary nodes and whether there are ongoing secondary transmissions. While a secondary node is transmitting secondary traffic, another secondary node relays the ongoing primary transmission to compensate for any interference caused by the ongoing secondary transmission. The results showed that a secondary system with full CSI can access



(a) TPSR based spectrum leasing scheme



(b) DF based spectrum leasing scheme

Figure 2.7: Spectrum Leasing Schemes (© [2013] IEEE [6]).

the spectrum without degrading the primary system performance, under the proposed spectrum leasing with secondary user selection scheme.

## **Chapter 3**

# **Secondary VoIP Capacity in Opportunistic Spectrum Access Networks with Friendly Scheduling (Constant Transmission Rate)**

### **3.1 Introduction**

Conventional cognitive radio typically assumes that the primary network is unchanged. However, in some situations the primary network operator may wish to accommodate secondary user access through informal or formalized relationships. Recently, *secondary user friendliness* has been addressed [84] that considers the option of having the primary network utilize its spectrum in a way that is more compatible with potential future secondary networks. Reference [84] proposes changes to the scheduling algorithm used at the primary

user stations so that the maximum throughput of secondary users can be improved without affecting primary network capacity. This is accomplished by deferring primary transmissions when channel conditions are poor and by sending them when channel quality has improved. This improves the primary network channel use, thus increasing spectrum hole availability for the secondary network.

In this chapter we take a different approach to secondary user friendliness based on primary basestation scheduling. Our approach is to minimize the change that would be required at the primary user basestation. Friendliness is measured as the number of secondary VoIP connections that can be obtained and is achieved by having the primary network shape its residual capacity subject to satisfying its own packet deadline constraints. We first develop an offline scheduler that can maximize friendliness over finite time intervals. Two online scheduling algorithms are then introduced that provide various levels of real-time friendliness. In the first algorithm the primary network shapes its residual capacity using an approach motivated by a conventional token bucket. The second algorithm uses virtual secondary VoIP calls and applies real-time scheduling to both real primary traffic and virtual secondary traffic. An issue with the second algorithm is that the primary network is not aware of the vocoding capabilities of the secondary users. Simulation results are presented that show the degree to which secondary user VoIP capacity can be improved compared to that of a conventional non-friendly scheduler. Analytic results are also presented that are based on a synchronous TDMA (STDMA) message delay model that can be used to approximate secondary user friendliness using primary traffic descriptors.

The remainder of the chapter is organized as follows. In Section 3.2 a brief overview is first given of related work. This includes work that proposes QoS-aware cognitive media access control protocols and recent results that consider cognitive secondary network VoIP

capacity. In Section 3.3 we then give a detailed description of our system model. Section 3.4 then formulates an optimum offline schedule for user friendliness when the secondary users are transmitting VoIP traffic. A formulation of this schedule is then given that can be efficiently solved using a minimum cost flow graph construction. In Section 3.5, online algorithms are introduced for secondary friendliness. This includes approaches based on adaptive token bucket spectral hole shaping and virtual node scheduling. Examples of results for secondary user friendliness are then given in Section 3.6. A wide variety of results are presented that include the effects of different secondary user vocoding capabilities on the secondary user capacity. An STDMA message delay model is developed in Appendix A, which can approximate secondary user friendliness and can be used to help validate the simulation model results. Finally, in Section 3.7 the chapter presents the conclusions of our work.

## 3.2 Related Work

As cognitive networks evolve, secondary network access will initially involve services with low quality-of-service (QoS) requirements, such as email, web browsing and other best-effort types of traffic [97]. Protocols for supporting non-real-time services that enforce minimum bit rate QoS constraints for secondary flows are described in [40]. This work focuses on bandwidth utilization and flow fairness. In [41] the approach taken is to minimize secondary network power consumption, subject to satisfying minimum target data rates for secondary users. The possibility of real-time applications with more stringent QoS requirements depends on the traffic characteristics of the primary network. In the long-term however, supporting QoS in the secondary network is desirable and recent work has included QoS-aware analyses for specific purposes [38] [42]. Cognitive radio protocols

have also been studied that support real-time services with minimum bit rate and delay constraints for secondary flows [49] [75]. In addition to this work, QoS-aware scheduling for real-time traffic that assesses the voice capacity of secondary users has been considered [4]. In this study, two control mechanisms are proposed that support cognitive voice services in the presence of primary users, and a voice capacity analysis was provided. Other work that considers secondary user voice capacity includes [59] and [63].

All of the work discussed above focuses on supporting real-time services in the secondary network. However, to the best of our knowledge, no previous work has considered modifications on the primary side to provide friendliness to secondary real-time services. The idea of secondary user friendliness has been considered in [84], where a secondary user friendly scheduler was proposed for the primary users. In this work, friendliness is proposed for best effort secondary traffic and is a bi-product of improved primary network bandwidth usage when there is sufficient primary traffic delay tolerance.

### **3.3 System Model**

Our goal is to design a secondary user friendly and QoS-aware scheduler for the primary basestation. We are interested in secondary user friendliness from the point of view of real-time VoIP traffic capacity in the secondary network. It is assumed that the primary basestation is unaware of the secondary users and their opportunistic use of the channel [11]. For a given primary network traffic scenario, friendliness is defined to be the maximum number of secondary VoIP traffic flows that can co-exist with the primary traffic, subject to typical VoIP QoS constraints. This definition is similar to that considered in the cognitive voice capacity analysis work discussed in Section 2.4.2. In this case however, scheduling is done at the primary basestation in an effort to improve secondary network VoIP capacity. This

must be done in a way that is consistent with the primary network's QoS traffic guarantees.

We assume that the primary users communicate with their basestation using a demand TDMA protocol. The time axis is divided into periodic superframe intervals, each of which consists of a fixed number of time slots used for packet data. An example of this is shown in Figure 3.1. Our focus is on the downlink communication between the basestation and the primary users, however, the ideas are also applicable for uplink communication. As is normally the case, the basestation transmits to its users in time slots on the downlink as shown in the figure.

We assume a secondary basestation that is monitoring the primary channel and supporting downlink VoIP transmission for its users. It is also assumed that the secondary basestation and users are within the full coverage range of the primary network basestation. In this case the secondary basestation can have a highly reliable spectrum sensing capability, and error free spectrum sensing is assumed [4] [63]. Accordingly, secondary transmissions can be transmitted into unused primary network time slots that are sensed to be idle. An example of this is shown in the third time slot of Figure 3.1. Secondary transmission is accomplished by carrier-sensing at the start of each time slot for a short time period that is sufficiently long to detect if the slot is occupied. A single time-invariant channel model is assumed where the stations use power control [4] [63] and where the packet size is fixed for both primary and secondary traffic flows. It is assumed that each primary packet can be serviced in one time slot. However, the number of secondary packets that can be carried by a single time slot depends on the slot size as well as the VoIP codec type and configuration used.

The case is considered where the primary network is transmitting multi-packet message data on the channel being considered. It is assumed that each message arriving to the

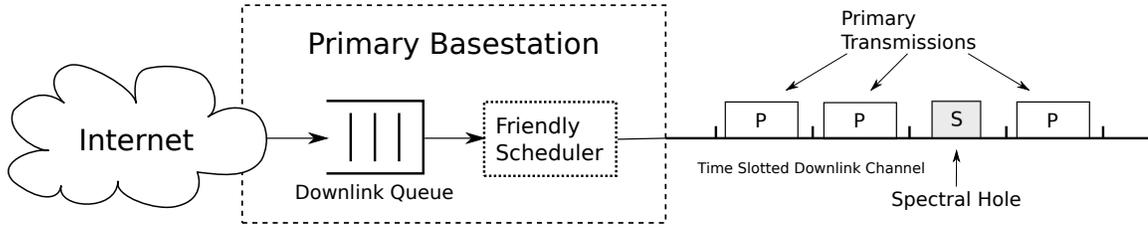


Figure 3.1: Friendly Basestation Scheduling Example. Primary traffic arrives to the basestation and is placed in the downlink queue (DLQ). The friendly scheduler (FS) transmits these packets subject to deadline constraints. Secondary users are free to monitor the downlink channel and transmit into unused time slots. Friendliness is measured by the number of secondary VoIP connections that can be supported for a given primary traffic scenario.

basestation is assigned a transmission deadline, which translates into deadlines for the corresponding packets. Note that there are many ways of assigning deadlines to the primary messages and in this development, we assume that this can be done arbitrarily. However, in the results in this thesis, deadlines are obtained based on a packet excess delay constraint. Secondary VoIP packets are assumed to arrive with a constant packetization interval and must be transmitted subject to delay/jitter constraints.

VoIP performance can be assessed in different ways. Mouth-to-ear measurements are sometimes used, including the R-factor [98], which reflects the codec type, delay, jitter and packet loss achieved. However, mouth-to-ear measurements are more commonly applied on an end-to-end basis. For a single one-way VoIP link, as considered in this thesis, conventional measures of delay and jitter are also commonly used. Our system model considers outgoing secondary VoIP links that use spectrum holes for transmission, i.e., full round-trip VoIP calls are not considered. For these reasons, delay and jitter requirements are used for modelling the QoS constraints of the secondary VoIP calls. The inputs and outputs of our scheduling problem are more formally defined as follows.

**INPUT:** Packet scheduling is to occur over a set of contiguous time slots, denoted by  $\mathcal{T}$ .

We define  $\mathcal{K}_p$  to be the set of  $K_p$  packets in a given primary traffic sample function. Every primary packet,  $i \in \mathcal{K}_p$ , has an arrival time and a delay constraint, and we define

$$\mathcal{T}_p(i) = \{ t \mid \text{time slot } t \text{ satisfies the QoS constraint for primary packet } i. \} \quad (3.1)$$

i.e.,  $\mathcal{T}_p(i) \subset \mathcal{T}$  is the set of time slots, any one of which would satisfy packet  $i$ 's QoS constraints if assigned to that packet. We also assume that a fraction of primary packets,  $\epsilon$ , can be dropped with no scheduling when their delay constraints are violated. Clearly,  $\epsilon = 0$  corresponds to the case that requires that all primary delay constraints are met.

In a similar way, we define a candidate set of  $F_s$  secondary flows given by  $\mathcal{F}_s$ , where for each  $m \in \mathcal{F}_s$  we define  $\mathcal{K}_s(m)$ , to be the set of  $K_s(m)$  VoIP packets belonging to flow  $m$ . Every secondary packet  $n \in \mathcal{K}_s(m)$  has an arrival time and delay/jitter requirement. We denote secondary flow/packet  $(m, n)$  for all  $m \in \mathcal{F}_s, n \in \mathcal{K}_s(m)$  and define

$$\mathcal{T}_s(m, n) = \{ t \mid \text{time slot } t \text{ satisfies the QoS constraint for secondary packet } (m, n) \} \quad (3.2)$$

i.e.,  $\mathcal{T}_s(m, n) \subset \mathcal{T}$  is the set of conforming time slots for VoIP packet  $n$  in flow  $m$ . Also, we define  $\beta$  to be the fraction of packets in each VoIP flow that can be dropped with no scheduling.

**OUTPUT:** The objective is for the primary network basestation to schedule packet transmission subject to meeting its own message deadline constraints and such that the

number of conforming voice flows,  $F_s$ , can be maximized. More formally, we define

$$x_p(i, t) = \begin{cases} 1 & \text{if primary packet } i \text{ is transmitted in time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (3.3)$$

for all  $i \in \mathcal{K}_p$  and  $t \in \mathcal{T}$ . Similarly we define

$$x_s(m, n, t) = \begin{cases} 1 & \text{if secondary packet } (m, n) \text{ is transmitted in time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

for all  $m \in \mathcal{F}_s$ ,  $n \in \mathcal{K}_s(m)$  and  $t \in \mathcal{T}$ . Given these definitions and the above inputs, the objective for the primary network is to assign values for  $x_p(i, t)$  that satisfy its own packet deadline constraints. *A friendly scheduler will also try to do this in a way that best accommodates the secondary user traffic, i.e., maximizes  $F_s$ .*

A friendly scheduler can generate either *offline* or *online* schedules. An *offline scheduler* is provided with the complete set of input sample functions for both the primary and secondary packets (i.e., for a given set of  $F_s$  VoIP flows), which it then uses to generate an offline schedule. The information known a priori includes all primary and secondary packet time slot conforming sets,  $\mathcal{T}_p(i)$  and  $\mathcal{T}_s(m, n)$  for all  $i \in \mathcal{K}_p$  and  $m \in \mathcal{F}_s, n \in \mathcal{K}_s(m)$ . An optimal offline schedule can be used to compute an upper bound on friendliness, as shown in Section 3.4. In contrast to offline scheduling, an *online scheduler* is provided with the primary packet inputs in real time and it must make causal downlink time slot assignments. In this case, neither  $F_s$  nor any of the VoIP packet arrival details are known to the basestation. In Section 3.5, online scheduling algorithms are discussed.

The STDMA message delay model presented in Appendix A gives an estimation of

the secondary VoIP capacity based solely on primary traffic descriptors and the secondary VoIP codec used. Using this information and with no input traffic traces, this analysis can approximate the achievable secondary VoIP capacity for a given primary traffic source. Before proposing the online schedulers and the analytic model, we first derive an optimal offline scheduler that maximizes the number of secondary VoIP flows that can be scheduled for given sets of input sample functions. This is introduced in the next section and is used for comparisons with the online schedulers.

### **3.4 Friendliness Upper Bound**

In this section we formulate an optimization that can be used to find an upper bound on the maximum number of real-time secondary flows that can access the primary network subject to protecting the QoS requirements of both the primary and secondary networks. The bound is obtained by taking a complete time window of primary inputs and determining if a given set of secondary flows can be accommodated. For this reason, the result gives a bound that in general will not be achievable by any causal scheduling algorithm.

To simplify the exposition, we assume that a time slot can accommodate only a single VoIP packet. It is easy to generalize this restriction in the offline schedule formulation and allow multiple VoIP packets per time slot, if desired. In Section 3.6, results are given using VoIP vocoders where multiple secondary packets can occupy a single time slot. We first derive an optimal offline schedule using an integer linear program whose solution has worse-case exponential time complexity. A minimum cost flow graph representation of the problem is then presented that derives the optimum schedule in time complexity that is polynomial in the number of time slots.

### 3.4.1 Optimum Offline Scheduling

The optimum offline friendliness is obtained using an optimization that is formulated as a feasibility problem. It is solved for the provided set of  $F_s$  secondary VoIP calls, i.e., assignments are made for  $x_s(m, n, t)$  and  $x_p(i, t)$  defined in Section 3.3. The bound is obtained by increasing the number of flows in  $\mathcal{F}_s$  until an infeasible solution is reached, i.e., the optimization is solved for a predetermined set of secondary flows,  $\mathcal{F}_s$  and the primary traffic, subject to all QoS constraints. If a feasible solution is found, then a new VoIP flow is added to  $\mathcal{F}_s$ , and the test is repeated. Eventually, the maximum level of friendliness is determined once the optimization reaches infeasibility. Using the definitions from Section 3.3, the feasibility test, denoted by OFFLINE–OPT, is as follows.

$$\text{minimize } 0 \quad (\text{OFFLINE–OPT})$$

subject to

$$\sum_{i \in \mathcal{K}_p} x_p(i, t) + \sum_{m \in \mathcal{F}_s} \sum_{n \in \mathcal{K}_s(m)} x_s(m, n, t) \leq 1, \quad \forall t \in \mathcal{T} \quad (3.5)$$

$$\sum_{t \in \mathcal{T}} x_p(i, t) \leq 1, \quad \forall i \in \mathcal{K}_p \quad (3.6)$$

$$\sum_{t \in \mathcal{T}} x_s(m, n, t) \leq 1, \quad \forall m \in \mathcal{F}_s, n \in \mathcal{K}_s(m) \quad (3.7)$$

$$\sum_{i \in \mathcal{K}_p} \sum_{t \in \mathcal{T}_p(i)} x_p(i, t) \geq (1 - \epsilon)K_p \quad (3.8)$$

$$\sum_{n \in \mathcal{K}_s(m)} \sum_{t \in \mathcal{T}_s(m, n)} x_s(m, n, t) \geq (1 - \beta)K_s(m), \quad \forall m \in \mathcal{F}_s \quad (3.9)$$

$$x_p(i, t) \in \{0, 1\}, \quad \forall i \in \mathcal{K}_p, t \in \mathcal{T} \quad (3.10)$$

$$x_s(m, n, t) \in \{0, 1\}, \quad \forall m \in \mathcal{F}_s, n \in \mathcal{K}_s(m), t \in \mathcal{T} \quad (3.11)$$

The constant objective in OFFLINE–OPT constructs a feasibility test for the input data using the defined constraints. Constraint (3.5) enforces the single channel assumption, where a single time slot can carry only one primary or secondary packet. Constraints (3.6) and (3.7) ensure that a given primary or secondary packet can be either scheduled once or not at all. Constraint (3.8) preserves the QoS constraints of the primary packets by ensuring that no more than the maximum allowable drop fraction,  $\epsilon$ , do not meet their QoS requirements. Similarly, constraint (3.9) ensures the QoS per flow guarantees for secondary packets. Constraints (3.10) and (3.11) define  $x_p(i, t)$  and  $x_s(m, n, t)$  as binary decision variables.

The constant objective in OFFLINE–OPT is used to test for the feasibility of a given set of VoIP calls. If a solution exists, then the system can satisfy all of the primary and secondary packet constraints for the current set of VoIP calls. By increasing the number of calls, eventually the optimization will become infeasible, indicating that the VoIP call capacity has been exceeded.

The ILP formulated above has worse-case exponential complexity. In the next section, however, we show that it can be solved in time that is polynomial in the number of time slots. This is done using a minimum cost flow graph construction.

### Minimum Cost Flow Graph Formulation

The ILP presented in OFFLINE–OPT can be solved in time complexity that is polynomial in the number of time slots using a minimum cost flow graph formulation. A flow graph,  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , is defined by a set,  $\mathcal{V}$ , of vertices (nodes) and a set,  $\mathcal{E}$ , of directed edges (arcs) connecting the nodes [99]. For each edge  $(i, j) \in \mathcal{E}$ , there is a capacity  $u_{i,j}$  that constrains the maximum flow on the edge and an associated cost,  $c_{i,j}$ , which denotes the cost per unit

flow along this edge. Each vertex  $v \in \mathcal{V}$  has an associated number  $b_v$  that represents the supply/demand of the vertex. If  $b_v > 0$ , the node is a supply node; if  $b_v < 0$ , node  $v$  is a sink node, and if  $b_v = 0$ , then node  $v$  is a transshipment node.

The OFFLINE–OPT program can be viewed as a Minimum Cost Flow Problem [99] and is shown in the flow graph of Figure 3.2. To simplify the description, the figure shows only a single secondary flow. For multiple secondary flows, the graph is extended for every added flow by replicating the part shown in the figure that represents the single secondary data unit flows and nodes, with corresponding scheduling and dropping edges added. The details of the graph construction are discussed below.

The flow enters and exits the graph at nodes  $S$  (Source) and  $D$  (Destination), respectively. The input flow represents the integer total of all primary and secondary packets, given by  $(K_p + \sum_{m \in \mathcal{F}_s} K_s(m))$  where  $K_p$  and  $K_s(m)$  represent the cardinality of the sets  $\mathcal{K}_p$  and  $\mathcal{K}_s(m)$ , respectively. The input flow is then split into primary and secondary packet flows as shown. A maximum fraction,  $\epsilon$ , of the input primary flow is allowed to be dropped, as previously discussed. This is shown by the  $(\epsilon K_p, 0)$  edge. Note that the  $\epsilon K_p$  product must be integral, which is easy to ensure when  $K_p$  is large, as is always the case. However, the full primary flow may still be scheduled if it is feasible. This is permitted by the two  $(K_p, 0)$  edges. In the same fashion, for every secondary flow  $m \in \mathcal{F}_s$ , there is a node in the first column that ensures that a maximum fraction,  $\beta$ , of packets belonging to the set  $\mathcal{K}_s(m)$  are allowed to be dropped. As in the primary flow case, dropped packets from the secondary flows proceed directly to the Destination node with no scheduling.

In Figure 3.2, the Packets column includes a node for each primary and secondary packet, and the Time Slots column contains a graph node for each time slot. *Each packet node has an edge connecting it to all the time slot nodes that satisfy the QoS requirements*

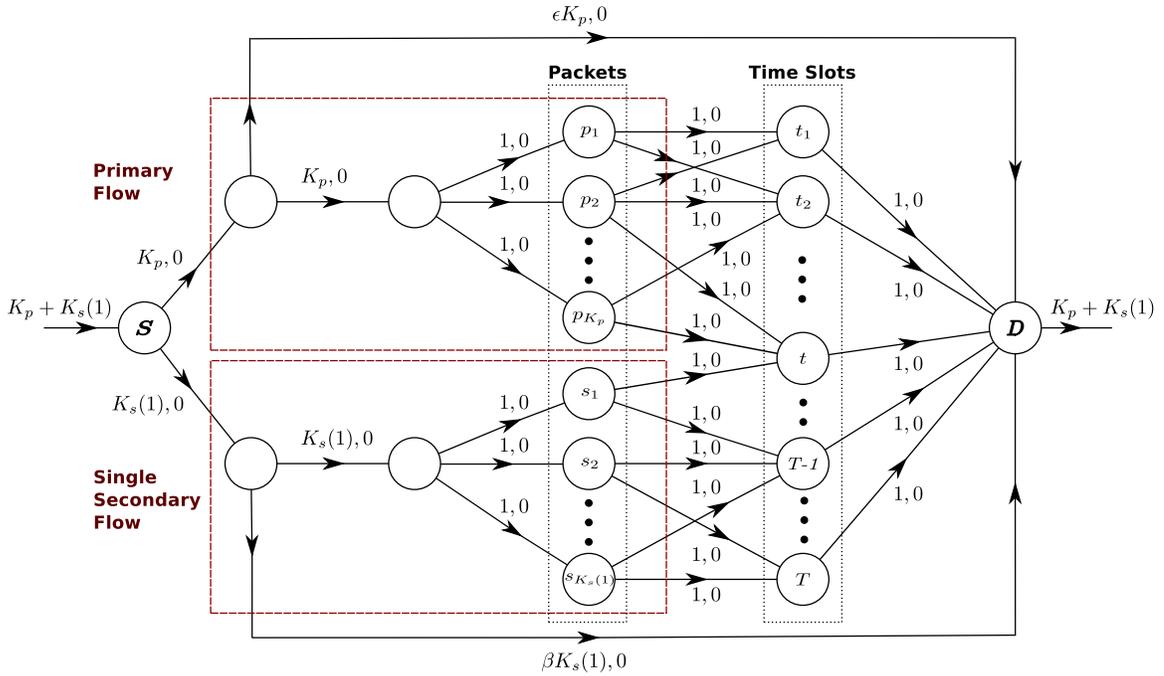


Figure 3.2: Offline Minimum Cost Flow Graph Scheduler. Each edge  $(i, j)$  is labeled with an ordered pair  $(u_{i,j}, c_{i,j})$  representing the capacity and cost of this edge, respectively. The input and output links have flow capacity of  $(K_p + K_s)$  with zero cost.

of that packet, i.e., the packet to time slot edges are taken from the entries in the sets  $\mathcal{T}_p(i)$  and  $\mathcal{T}_s(m, n)$  for the primary and secondary packets, respectively. A selection of one of these edges (time slots) will satisfy the QoS requirements for the packet being considered.

The capacities and costs for the last three columns of edges are  $(1, 0)$ . This ensures that at most one unit of flow will proceed along any of these edges. In the final edge column, for example, this constraint ensures that at most one packet will be assigned to a given time slot. The cost for using all these edges is zero, since we are interested in solving a feasibility test, to determine if a feasible schedule exists for the given set of inputs.

When the flow graph is solved, the Integrality Property Theorem is invoked (e.g., Theorem 9.10 in K. Ahuja et. al. [99]), which states that *If all edge capacities and supplies/demands of nodes are integer, the Minimum Cost Flow problem always has an integer minimum cost flow.* As a result, the flow between the packet and time slot nodes is integer. Coupled with the fact that the edge capacities are 1, the resulting flows are binary variables, which gives us a valid schedule, i.e., the values for  $x_p(i, t)$  and  $x_s(m, n, t)$  in OFFLINE-OPT will be binary, as required. This formulation gives a way of computing the optimum offline friendliness by solving the minimum cost flow graph for each candidate set of VoIP flows as discussed above. This can be efficiently done in time complexity that is polynomial in the number of time slots.

### 3.5 Online Secondary Friendly Scheduling

In this section we propose two online secondary friendly schedulers. The algorithms operate at the primary basestation and make scheduling decisions at each time slot based on the pending primary traffic backlog and its QoS requirements.

The first algorithm, *Adaptive Token Bucket Spectrum Hole Shaping Scheduler (TB)*, achieves secondary VoIP friendliness by attempting to create periodically spaced spectrum holes for potential secondary voice transmissions. This is done by shaping the primary traffic to prevent primary bursts (messages) from occupying the channel for periods of time that would be inconsistent with secondary QoS requirements. For this purpose we use a simple token bucket where the token rate is dynamically set when new primary messages arrive.

The second algorithm, *Virtual Node Scheduler (VN)*, uses a real-time scheduler to coordinate both *real* primary traffic and *virtual* secondary VoIP flows. Ties in contention are

resolved in favor of the primary traffic. The performance of the virtual node scheduler depends on the differences between the assumed virtual VoIP codec configuration and the real VoIP codec configuration. Both perfect codec match and different cases of codec mismatch between the real and virtual codecs are considered. Details of the two schedulers are introduced in the following two sections.

### **3.5.1 Adaptive Token Bucket Spectrum Hole Shaping Scheduler (TB)**

The approach taken is to shape primary channel usage using a simple variant of the well-known token/leaky bucket scheme that is used to regulate traffic burstiness [100]. In a conventional token bucket, tokens are generated every  $1/r$  seconds and stored in a bucket that has a capacity of  $W$ . A token must be removed from the bucket in order for a packet to be transmitted. If the number of tokens in the bucket exceeds  $W$ , no more tokens are generated. It is well known that two parameters affect token bucket performance,  $(r, W)$ , which defines the maximum sustainable data rate and the maximum burst size that can be allowed into the network. In our case this applies to the primary packets as shown in Figure 3.3.

The primary traffic is message based with deadline requirements, and our objective is to create VoIP compatible spectral holes. For our purposes we set the bucket capacity  $W = 1$  and set a different value for the token rate  $r$  upon every primary message arrival. The value of  $r$  is selected to be the lowest token rate that satisfies the QoS requirements of the current pending primary traffic. Setting  $W = 1$  spaces out primary packet transmissions, which tends to create friendly spectral holes to VoIP users. We have found that when  $W > 1$ , larger primary packet bursts tend to decrease friendliness. Shaping the primary traffic using a token bucket with  $W > 1$  reduces the primary traffic friendliness significantly or even

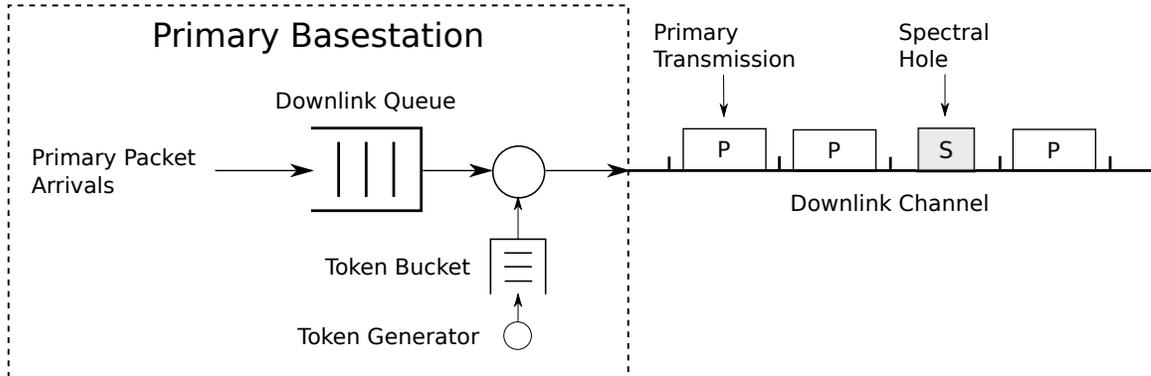


Figure 3.3: Token Bucket Spectral Hole Shaping. To transmit, a packet must obtain a token from the token bucket. A new token is generated every  $1/r$  seconds, where  $r$  is the set input rate, as long as the number of tokens does not exceed a given threshold.

eliminates it in some situations. Because higher values of  $W$  allows primary traffic bursts to be transmitted, this tends to block the channel for consecutive time periods, making it more difficult for VoIP calls to coexist.

Accordingly, we define a binary variable  $B$  that is equal to 1 if there is an available token and 0 otherwise. After setting  $W = 1$ , the token rate  $r$  is adjusted to achieve the most friendly primary traffic output. However, as the burstiness of the primary traffic increases, the adaptive TB scheduler can assign very low token rate values at light loads, which may lead to infeasibility of the solution later when larger messages arrive. A proposed solution to this problem is to set a minimum token rate threshold,  $r_{min}$ , where the primary traffic mean arrival rate is a good candidate for this value. However, different values of the minimum token rate threshold are also considered in Section 3.6.

A more formal description of the scheduler is shown in Algorithm 1. In steps 2 to 5, the current token rate value,  $r(t)$ , is recomputed every time a new primary message arrives. Step 3 defines the current new scheduling window time frame  $\mathcal{T}_{SW}$ , which starts at the current time slot  $t$  and ends at the farthest known future deadline  $t_d$ . Step 4 changes the

**Algorithm 1** TB Algorithm

$\mathcal{T}$  : set of all time slots  
 $\mathcal{T}_{SW}$  : set of time slots in the current scheduling window, redefined upon new message arrivals,  $\mathcal{T}_{SW} \subset \mathcal{T}$   
 $\mathcal{K}_p(t)$  : primary packet backlog at time slot  $t$   
 $e(t)$  : index of the primary packet with the earliest deadline among  $\mathcal{K}_p(t)$  at time slot  $t$   
 $\mathcal{R}$  : set of possible values of token generation rate, i.e.,  $\mathcal{R} = \{1, \frac{1}{2}, \frac{1}{3}, \dots, r_{min}\}$   
 $r(t)$  : current value of token generation rate at time slot  $t$   
 $B(t)$  : binary current bucket level at time slot  $t$ , i.e.,  $B(t) = 1$  if a token exists, 0 otherwise

```

1: for all  $t \in \mathcal{T}$  do
2:   if a new primary message(s) arrive(s) then
3:     Define  $\mathcal{T}_{SW} = \{t, t + 1, \dots, t_d\}$ , where  $t_d$  is the largest primary packet deadline.
4:     Set  $r(t)$  to the minimum token rate value  $\in \mathcal{R}$ , that satisfies the QoS of  $\mathcal{K}_p(t)$  packets
       over  $\mathcal{T}_{SW}$ .
5:   end if
6:   TB-GENERATE-TOKEN( $t, r(t), B(t)$ )
7:   if pending primary packets exist and a token is ready, i.e.,  $|\mathcal{K}_p(t)| > 0$  &  $B(t) = 1$  then
8:     Send the most urgent primary packet,  $e(t)$ .
9:     Update current bucket level  $B(t)$ , i.e.,  $B(t) = 0$ .
10:  end if
11: end for
12: function TB-GENERATE-TOKEN( $t, r(t), B(t)$ )
13:   if bucket is empty and it is time for a new token, i.e.,  $B(t) = 0$  &  $t \bmod \frac{1}{r(t)} = 0$  then
14:     Generate a new token, i.e.,  $B(t) = 1$ 
15:   end if
16:   return  $B(t)$ 
17: end function
  
```

current value of the token rate,  $r(t)$ , to the minimum possible value greater than  $r_{min}$  that can satisfy the delay requirements for the current pending primary packet backlog  $\mathcal{K}_p(t)$ , as of time slot  $t$ .

In step 6, the TB-GENERATE-TOKEN function call is responsible for new token generation. The TB-GENERATE-TOKEN function is declared in steps 12 to 17 and accepts the current token rate  $r(t)$ , the current time slot  $t$  and the current bucket level  $B(t)$ . It then returns an updated bucket level that reflects whether a new token has been generated. In

steps 7 to 10, the current token rate value  $r(t)$ , is used to shape the current primary backlog. If a token is available and at least one pending primary packet exists, the most urgent primary packet, denoted by  $e(t)$  is transmitted and the current bucket level  $B(t)$  is updated. Some obvious implementation details are not shown in the algorithm description for the sake of clarity. Essentially,  $\mathcal{K}_p(t)$  and  $e(t)$  are updated on a per time slot basis to reflect online arrivals and departures of primary packets. Moreover, at every time slot  $t$ , each of  $B(t)$  and  $r(t)$  are initialized to the bucket level and the token rate set at the previous time slot  $t - 1$ , respectively, i.e.,  $B(t) \leftarrow B(t - 1)$  and  $r(t) \leftarrow r(t - 1)$ .

### 3.5.2 Virtual Node Scheduler (VN)

The virtual node scheduler takes a different approach by scheduling both *real* primary packet transmissions and *virtual* secondary VoIP transmissions. An example of this is given in Figure 3.4. The real primary packets arrive to the primary downlink queue as shown. Similarly, (simulated) virtual secondary packet arrivals arrive to the virtual VoIP queues, one for each assumed secondary VoIP connection. The virtual secondary transmissions do not exist of course, but merely serve to create free time slots that can be used by actual secondary users. The VoIP codec and associated parameters assumed for the virtual VoIP calls is referred to as the *virtual codec configuration*. Virtual secondary VoIP traffic is generated based on a CBR traffic model, where the VoIP packet size and packetization period is determined based on the virtual codec configuration used, with typical VoIP QoS constraints. Since both real primary traffic and virtual VoIP traffic have deadline constraints, a scheduler must mediate between the two. Specifically, an Earliest Deadline First (EDF) algorithm is used to schedule both primary traffic and virtual secondary traffic, with contention resolved in favor of the primary traffic.

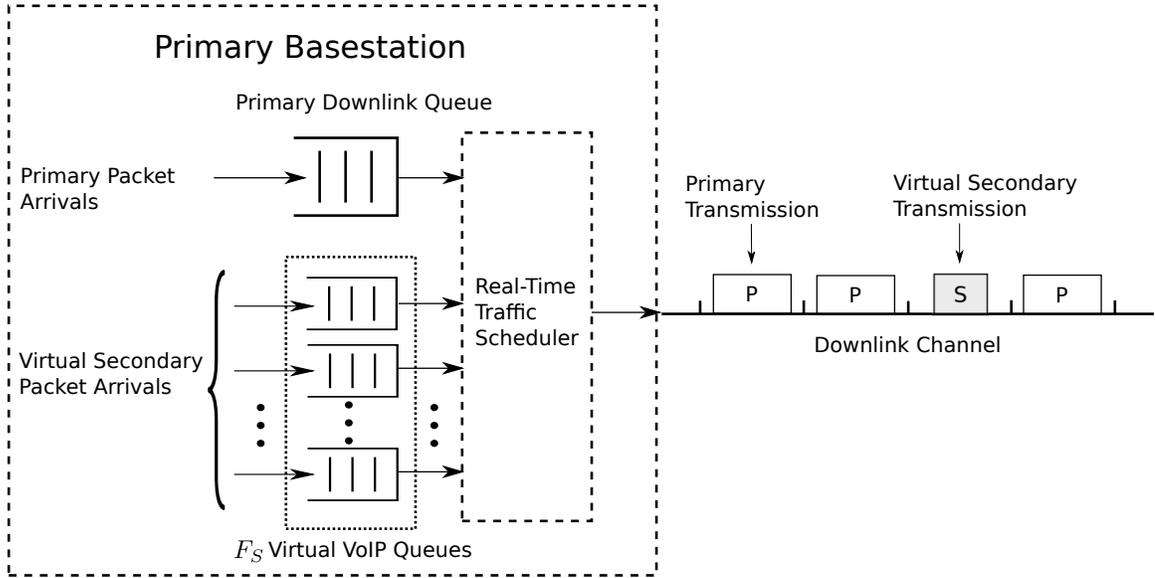


Figure 3.4: Virtual Node Online Friendly Scheduler. Primary traffic arrives to the basestation and is placed in the downlink queue (DLQ). The scheduler schedules real primary traffic and  $F_s$  virtual secondary VoIP calls, where contention is resolved in favour of primary traffic. Earliest Deadline First (EDF) is used for real-time traffic scheduling. Real Secondary VoIP calls accessing the channel will make best use of virtual secondary transmission slots (spectral holes).

A more formal description of the algorithm is shown in Algorithm 2. At time  $t$  it deals with two types of queues, the real primary traffic queue,  $Q_p(t)$  and the virtual secondary traffic queue,  $Q_s(t)$ , which carries aggregated traffic from the virtual secondary VoIP flows that belong to the set of virtual secondary flows, defined as  $\mathcal{F}_{vs}$ . Both queues are sorted with the packet having the earliest deadline at the head of the queue, denoted by  $HoQ_p(t)$  and  $HoQ_s(t)$  for the primary and secondary queues, respectively. At every time slot  $t$ , the algorithm (steps 2 to 6) performs a comparison between the deadlines of  $HoQ_p(t)$  and  $HoQ_s(t)$ . The head of queue packet with the earliest deadline will win the competition and be assigned to the current time slot,  $t$ . To prioritize the real primary traffic, ties are resolved in their favour. It is therefore guaranteed that the primary traffic is prioritized even in the

---

**Algorithm 2** VN Algorithm

---

$\mathcal{T}$  : set of all time slots  
 $\mathcal{F}_{vs}$  : set of virtual secondary flows  
 $Q_p(t)$  : current queue of primary packets at time slot  $t$  sorted with earliest deadline first (EDF)  
 $Q_s(t)$  : current *virtual* aggregated queue of secondary packets belonging to  $\mathcal{F}_{vs}$  flows at time slot  $t$  sorted with earliest deadline first (EDF)  
 $HoQ_p(t)$ : index of head of queue packet of  $Q_p(t)$   
 $HoQ_s(t)$ : index of head of queue packet of  $Q_s(t)$   
 $n_s$  : number of secondary packets that fit in a single time slot, which is calculated based on the *virtual* codec type and configuration used

```

1: for all  $t \in \mathcal{T}$  do
2:   if Deadline of primary packet  $HoQ_p(t) >$  Deadline of virtual secondary packet  $HoQ_s(t)$ 
   then
3:     Assign time slot  $t$  to the first  $n_s$  secondary packets of the virtual aggregated secondary
     queue  $Q_s(t)$ .
4:   else
5:     Send primary packet  $HoQ_p(t)$ .
6:   end if
7: end for

```

---

case of awaiting urgent packets when the deadline of both  $HoQ_p(t)$  and  $HoQ_s(t)$  is the current time slot,  $t$ . Since an urgent packet will miss its deadline if it is not transmitted immediately, in such cases the algorithm will resolve the tie by assigning the slot to the primary packet and dropping the virtual secondary packet. Essentially, both  $Q_p(t)$  and  $Q_s(t)$  are updated every time slot based on online arrivals and departures; this detail is not shown in the algorithm steps. The virtual secondary traffic scheduler follows the secondary schedule assigned at step 3, which, in turn, updates the  $Q_s(t)$  queue.

The virtual secondary voice capacity,  $F_{vs}^*$ , will be equal to the real secondary voice capacity,  $F_s^*$ , achieved when there is perfect codec match, i.e., when the virtual codec type and configuration assumed match those of the real secondary VoIP flows. However, in cases of VoIP codec mismatch, the real secondary voice capacity,  $F_s^*$ , achieved will be less than the virtual secondary voice capacity,  $F_{vs}^*$ , computed by the VN algorithm for the same

primary traffic traces. Both perfect codec match and different cases of codec mismatch between the real and virtual codecs are studied and presented in Section 3.6.

Both of the proposed online scheduling algorithms require no prior knowledge of the primary traffic arrival process distribution. However, as mentioned earlier, the TB algorithm tracks the mean arrival rate of the primary traffic, to set a lower bound on the token bucket rate used to shape the primary traffic. As explained in Section 3.5.1, this helps to ensure that the algorithm can satisfy primary traffic arrival constraints. This becomes more important as the burstiness of the primary traffic increases. The results presented in Section 3.6 show how the secondary friendliness level of a primary network is affected as the distribution of the primary traffic arrival process changes.

### 3.6 Simulation Model and Results

In this section we investigate the secondary voice capacity performance of the system under varying parameter assumptions. The main factor that affects potential secondary user friendliness is the willingness of the primary basestation to tolerate delay beyond that which would be obtained if no friendliness is permitted. Various definitions of delay tolerance are possible, however, in our results we define it relative to the non-friendly packet delay performance. Specifically, the deadline for a given primary packet is set to be the departure time that the primary packet would achieve in a non-friendly case plus a fixed maximum allowable excess delay,  $D_e$ . This is a simple definition that requires that packet delay is increased by at most  $D_e$  beyond the ideal, i.e., non-friendly case. Note that if the primary network is unwilling to suffer any change in performance, no matter how imperceptible it may be, then it would be very difficult to share the channel in a friendly fashion. This corresponds to the  $D_e = 0$  case. Note also that there is no primary throughput degradation. The only primary

performance degradation is in delay, i.e., the schedulers assign transmissions so that spectral holes are more VoIP favourable, subject to the primary network delay constraints. The results that we present show the friendliness for a given primary system throughput that must always be satisfied with finite mean delay.

Note that in the excess delay definition defined above, no two primary packets can have the same deadline. This is due to the fact that the deadline of a given primary packet is set to its departure time in a conventional non-friendly scheduler plus the excess delay setting. This means that even primary packets belonging to the same message, with the same arrival time, will have different packet deadlines since they have different departure times in a conventional non-friendly scheduler.

For simulation purposes, we assume a secondary basestation that is monitoring the primary channel and supporting downlink VoIP transmissions for its users. The method described in [4] is used to determine the secondary VoIP capacity. In this reference, to determine the maximum number of secondary flows subject to a per flow packet loss ratio, the system is repeatedly simulated for an increasing number of admitted secondary VoIP flows until the QoS requirements of any of the secondary flows is violated. The highest feasible number of secondary VoIP flows gives the secondary VoIP capacity. The VN algorithm is executed for an increasing number of admitted virtual secondary flows,  $F_{vs}$ , until the QoS requirements of any of the real primary traffic or virtual secondary flows are violated. The corresponding value then gives the virtual secondary voice capacity,  $F_{vs}^*$ . For simulation purposes, this is done over the entire simulation time, however for real online use of the VN scheduler, the virtual secondary VoIP capacity would be adaptively re-estimated upon every new primary message arrival, which converges to the same number over the simulation duration.

A secondary network is considered where both the secondary basestation and the secondary users are within the full coverage range of the primary network basestation. In this case the secondary basestation can have a highly reliable spectrum sensing capability, and error free spectrum sensing is assumed [4] [63]. The secondary basestation executes a QoS-aware packet scheduler that aims at satisfying the deadline and jitter requirements of the maximum number of secondary VoIP calls that can be supported. Earliest Deadline First (EDF) scheduling is used to transmit secondary VoIP packets based on deadline prioritization.

In our results, we assume both Poisson process and Pareto type 2 or Lomax distribution primary message arrivals. Pareto type 2 can generate values lower than the scale parameter, keeping a heavy tail behaviour, like the traditional Pareto distribution. The primary message size is fixed to 30 packets in length, with packet size set to 512 bytes. The secondary VoIP packet size is calculated as the summation of the VoIP payload size (which is based on the VoIP codec configuration used [101]), IP overhead (40 bytes) and L2 overhead (108 bytes for the IEEE 802.11 MAC). It is assumed that a slot size is equal to the primary packet size, so a single slot can carry one primary packet, while the number of secondary packets that fit in a single slot depends on the VoIP codec configuration used. For the simulation results presented below, the maximum packet loss ratio for primary traffic is set to 0.5%, while the maximum packet loss ratio for a secondary VoIP call is set to 3% [59] and all simulations are averaged over 10 runs. MATLAB was used for primary/secondary traffic generation and for implementing the TB Algorithm, VN algorithm and the FCFS non-friendly primary scheduler [102]. AMPL [103] was used for modelling the offline minimum cost flow graph formulation of the friendliness upper bound and the CPLEX optimizer [104] was used for solving it.

The online schedulers are first compared to the offline scheduling bound derived in Section 3.4 and a (non-friendly) conventional first-come-first-served (FCFS) scheduler [102] under varying primary traffic delay tolerance. The FCFS scheduler uses simple first-in-first-out (FIFO) scheduling [102], where packets are scheduled for transmission in their order of arrival. Hence, a FCFS primary scheduler would simply schedule a primary packet when it arrives and when a time slot becomes available. Results showing the performance of the VN and TB schedulers under varying VoIP codec configurations are included. We have also experimented with different values of the  $r_{min}$  value for the TB scheduler. Results are presented showing the effect of changing the tail shape parameter of the Pareto primary arrival distribution on the secondary VoIP capacity for both TB and VN algorithms. The effect of varying the QoS requirements of the secondary traffic is also studied.

### 3.6.1 Optimum Offline vs. Online Schedulers

In this section we consider the performance of the proposed algorithms with varying primary packet delay tolerance compared to the optimum offline scheduling bound derived in Section 3.4. We assume that primary messages arrive to the basestation according to a Poisson arrival process with a mean inter-arrival time of 100 ms (i.e., mean message arrival rate = 1/100 messages/ms), and since the message size is 30 packets, the mean packet arrival rate is on average 1/3 packets/ms. As for the secondary VoIP call QoS requirements, the delay and jitter requirements are set to 100 ms and 20 ms, respectively [105]. The secondary VoIP call codec configuration used is G711 with a 20 ms packetization period in Figure 3.5 and G711 with a 30 ms packetization period in Figure 3.6. For these secondary VoIP codec configurations, the channel slot size can carry only one secondary packet that satisfies the single packet per slot constraint (3.5), defined in the friendliness upper bound

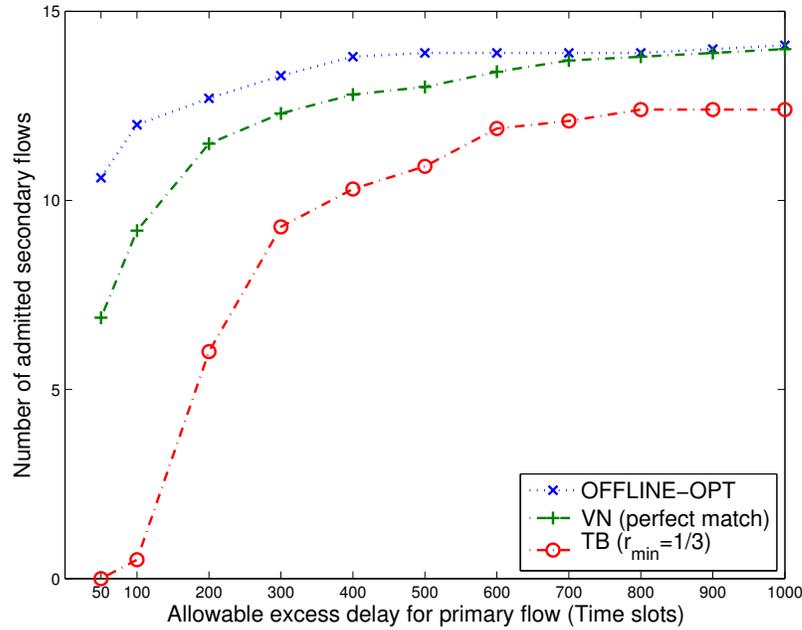


Figure 3.5: Secondary VoIP Capacity vs. Primary Delay Tolerance for OFFLINE-OPT, VN and TB Schedulers. The VN scheduler is in the perfect codec match case. G711-20 ms is used as the real codec configuration.

formulation. The value of the maximum allowable excess delay for the primary is varied to change the primary traffic delay tolerance.

In Figures 3.5 and 3.6, it is clear that the higher the delay tolerance of the primary traffic, the more secondary friendly it can be, since it can admit a larger number of secondary calls. As expected, the offline scheduler achieves the highest secondary VoIP capacity. Figures 3.5 and 3.6 also show the performance of the TB scheduler with the minimum token rate value set to the primary packet mean arrival rate, as well as the VN scheduler in the case of perfect codec match, i.e., the codec configuration of both virtual and real VoIP calls is the same. These figures show that the performance of the VN scheduler in the perfect VoIP codec match case is better than the TB scheduler, since the VN scheduler achieves a secondary VoIP capacity that is close to that of the upper bound. Note that the upper

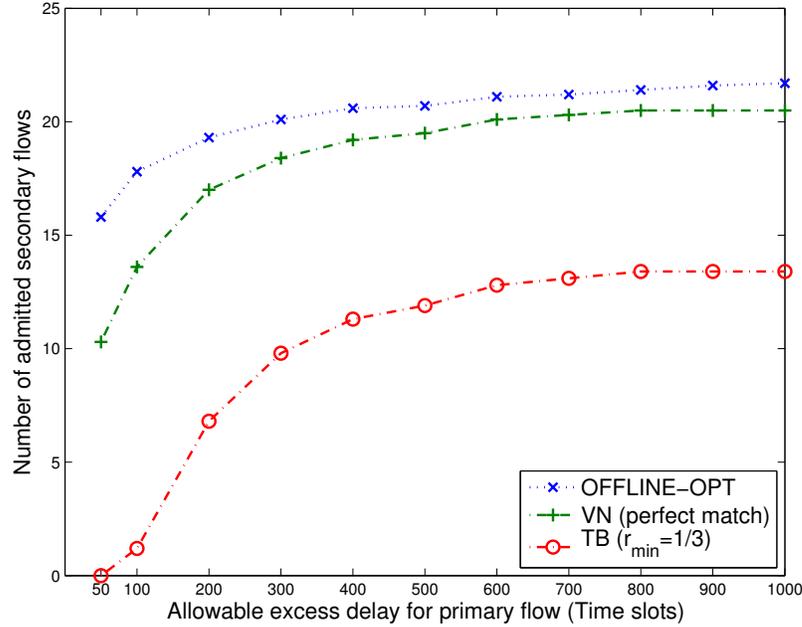


Figure 3.6: Secondary VoIP Capacity vs. Primary Delay Tolerance for OFFLINE-OPT, VN and TB Schedulers. The VN scheduler is in the perfect codec match case. G711-30 ms is used as the real codec configuration.

bound saturates at 14 VoIP calls in Figure 3.5 and at 21 VoIP calls in Figure 3.6, which is caused by the jitter requirement of the secondary calls. Moreover, it is clear that under tight packet delay constraints, the TB scheduler is unable to schedule a single secondary VoIP call, while the VN scheduler is able to admit about 6 secondary VoIP calls in Figure 3.5 and 10 secondary VoIP calls in Figure 3.6.

For the same traffic traces and simulation parameters, a conventional implementation that uses first-come-first-serve (FCFS) scheduling [102], achieves zero secondary VoIP capacity. For message based primary traffic the FCFS scheduler is non-friendly and cannot admit a single secondary VoIP call for different values of the primary message delay requirement. This is because it blocks the channel continuously until an entire message departs, thus violating the deadline and jitter requirements of potential secondary calls.

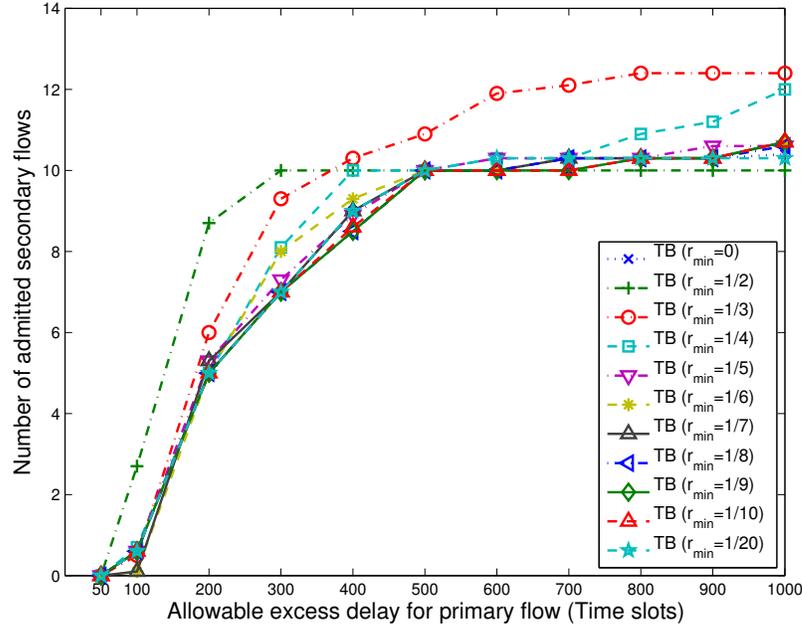


Figure 3.7: Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different values are used for  $r_{min}$ . G711-20 ms is used as the codec configuration.

Moreover, a FCFS scheduler doesn't account for the primary message delay requirement, since it schedules the packet departure directly after its arrival or at the next available slot, regardless of its deadline.

### 3.6.2 TB Scheduler: The Minimum Token Rate Threshold $r_{min}$

In this section the performance of the TB Scheduler is presented with varying primary packet delay tolerance, for different minimum token rate threshold values,  $r_{min}$ . The simulation parameters are the same as those used in Section 3.6.1.

Figures 3.7 and 3.8 show the effect of changing the minimum token generation rate threshold on the performance of the TB scheduler. Figures 3.7 and 3.8 use G711 as the secondary VoIP codec with the packetization period set to 20 ms and 30 ms, respectively. It is clearly shown in the figures that the primary mean packet arrival rate, i.e., 1/3, is a good

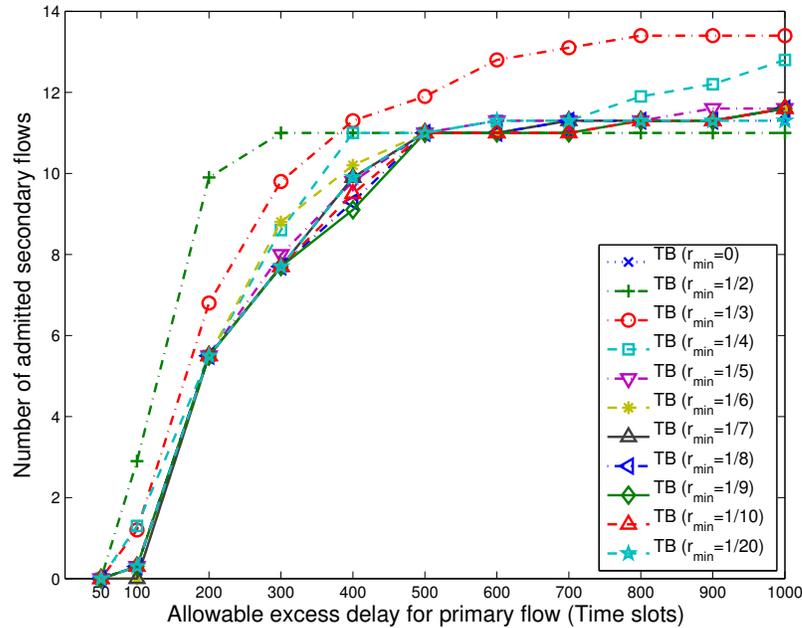


Figure 3.8: Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different values are used for  $r_{min}$ . G711-30 ms is used as the codec configuration.

candidate for  $r_{min}$ . Figures 3.7 and 3.8 show that when  $r_{min}$  is set to values less than  $1/3$ , the secondary VoIP capacity is reduced. This is because the scheduler makes more relaxed decisions, using low token rates for shaping primary packets at light primary traffic load. This, in turn, leads the scheduler to use a high token rate when several primary messages arrive in close time proximity. Since the TB scheduler dynamically adjusts the token rate upon each new message arrival, periods with high token rate decrease the overall secondary VoIP capacity that can be achieved. These results lead us to conclude that setting  $r_{min}$  close to the primary packet mean arrival rate results in good performance over a wide range of parameter values.

In all of our experiments, a default bucket size parameter of  $W = 1$  was used for the TB scheduler. Experiments have shown that the secondary VoIP capacity decreases monotonically with  $W$  for the reasons discussed in Section 3.5.1. For example, with  $r_{min} =$

1/3 in the G711-20ms case considered in Figures 3.5 and 3.7, at an excess delay of 300 ms, there is a 10% drop in secondary VoIP capacity when  $W$  is increased from 1 to 10. This reduction increases to over 70% when  $W = 20$ . When the excess delay is increased to 800, the reductions are 10% and 50%, for  $W = 10$  and  $W = 20$ , respectively. The corresponding reductions in capacity found for the case of G711-30ms are very similar. These, and other results confirm that  $W = 1$  is a good default bucket size.

### 3.6.3 VoIP Codec Configuration

In this section we present performance results for different secondary VoIP call codec configurations. The simulation parameters used are the same as those used in Section 3.6.1. In these results the virtual VoIP codec configuration for the VN scheduler in the codec mismatch case is set to G711 with a 20 ms packetization period.

Figures 3.9, 3.10 and 3.11 show the secondary VoIP capacity achieved by the TB scheduler, the VN scheduler in the perfect codec match case and the VN scheduler in the codec mismatch case, respectively, for different VoIP codec configurations. In the three figures it can be seen that the G711 codec with different packetization periods achieves the lowest secondary VoIP capacity, since it has the largest VoIP payload size (80, 160 and 240 for 10 ms, 20 ms and 30 ms packetization periods, respectively [101]). Hence the number of secondary VoIP packets that fit in a single slot is the least compared to other codec types. The G723 codec, which has a low bit rate with a 30 ms packetization period, achieves the largest VoIP capacity since it has a smaller VoIP payload size (20 bytes for 30 ms packetization [101]). As one would expect, the number of secondary VoIP calls allowed depends on the VoIP codec type and packetization period configured. Also, comparing the three figures for a given VoIP codec configuration, we can conclude that the performance of the

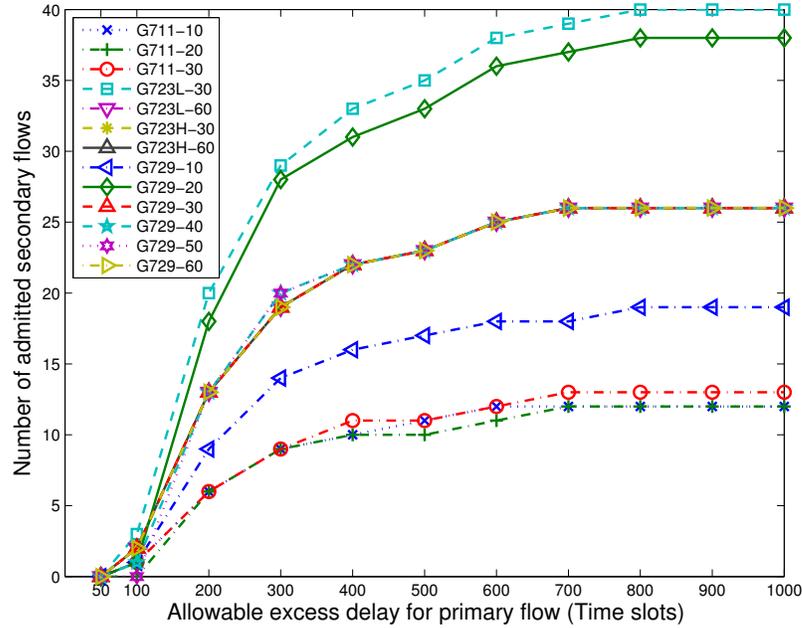


Figure 3.9: Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different VoIP codec configurations are considered.

VN scheduler in the perfect codec match case is generally better than its performance in the codec mismatch case (with G711-20 ms as the virtual codec used). However, both are still better than the performance of the TB scheduler for the same traffic traces.

There is a fair bit of statistical variation in some of the results that were generated, which is often seen in simulations where the time constants are quite different. In Figure 3.11 for example, at a primary excess delay value of 600 ms, a decrease in the achieved secondary VoIP capacity is seen for some of the VoIP codec configurations used. In this particular case, one of our input sample functions was somewhat atypical, however, most of the algorithms are affected similarly, and there is enough consistency to make reasonable comparisons.

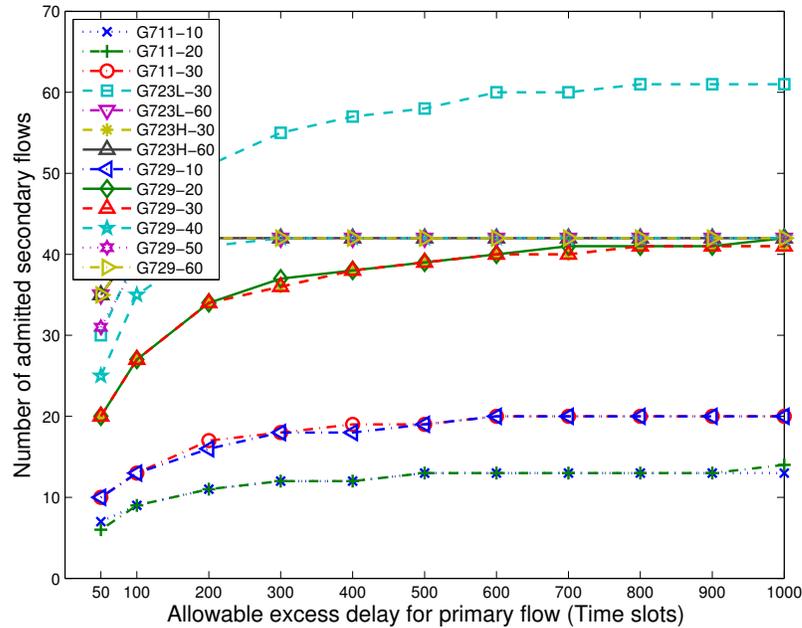


Figure 3.10: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VN Scheduler. Different VoIP codec configurations are considered (perfect codec match case).

### 3.6.4 VN Scheduler: Codec Match vs. Mismatch

In this section we use the same simulation parameters used in section 3.6.3 to investigate the performance of the VN scheduler in the case of codec mismatch. The codec configuration used for the real secondary VoIP call is G711 with a 30 ms packetization period.

Figure 3.12 shows the performance for different codec configurations of the virtual secondary VoIP calls compared to the performance of the TB scheduler and the offline bound. As expected, the offline scheduler achieves the best secondary VoIP capacity when compared to the online schedulers. Moreover, the VN scheduler in the perfect codec match case performs better than both the TB algorithm and the VN scheduler in the case of codec mismatch. It is also shown in Figure 3.12 that for some cases of codec mismatch, the VN scheduler performs worse than the TB scheduler, e.g., when the virtual codec is set to G729 with a 40, 50 or 60 ms packetization period. However, in other codec mismatch cases, e.g.,

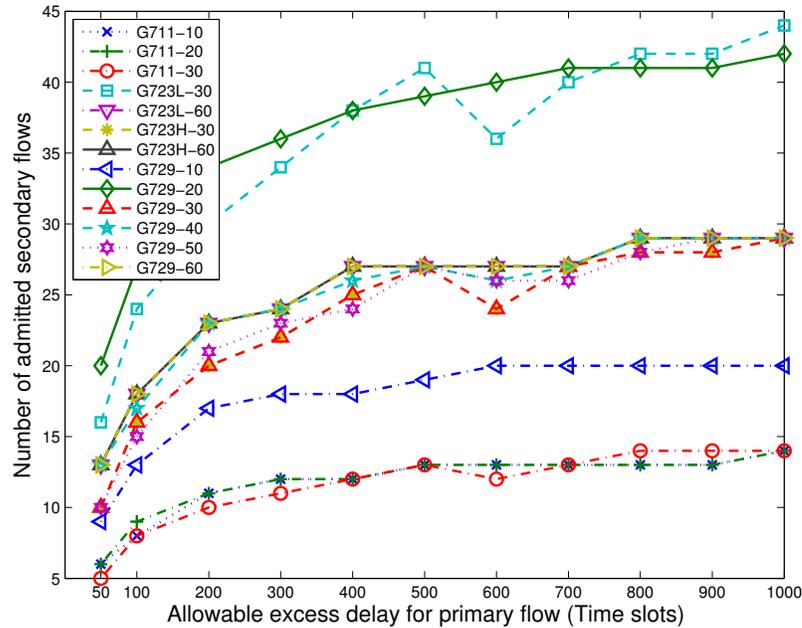


Figure 3.11: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VN Scheduler. Different VoIP codec configurations are considered (codec mismatch case (G711-20 ms as the virtual codec configuration)).

when the virtual codec configuration is set to G711 with a 20 ms packetization, the VN scheduler's performance is better than that of the TB scheduler.

We can conclude from these results that the performance of the VN scheduler depends on the selection of the codec configuration of the virtual secondary VoIP calls used in the VN algorithm and the codec configuration of the real secondary VoIP calls. In the case of codec mismatch, the VN algorithm may perform better or worse than the TB scheduler, depending on the parameter settings. However, in the case of perfect codec match, the VN is more friendly than the TB scheduler, but it is impractical to assume that the VN algorithm will always assume the same codec configuration used by potential secondary VoIP calls. One approach when using the VN scheduler is to assume a popular codec type such as G711 for the virtual codec with its default packetization period (20 ms) since it is supported by

most mobile devices. Another approach is to assume an adaptive VoIP codec module at the secondary nodes that tracks the activity of the primary users, then selects the VoIP codec that achieves the most efficient spectrum usage. Although this solution will achieve the performance of the VN scheduler in the perfect codec match case, it requires higher sophistication at the secondary nodes. An appropriate solution would be a combination of the two approaches, where the codec configuration of the virtual VoIP call used in the VN algorithm is set to G711 with 20 ms, while on the secondary side, the adaptive VoIP codec scheme would track the spectrum holes. In the recent literature, online scheduling algorithms were proposed to adapt the VoIP codec used in the secondary network based on the primary user activity [53] [52]. Reference [106] considers a system which controls VoIP call distortion by varying the VoIP source coding rate and other parameters in a downlink wireless system.

### **3.6.5 Heavy Tailed Primary Traffic with Different Tail Shape Parameter Values**

In this section we present results that assume a primary traffic arrival process based on the Pareto distribution. The Pareto distribution is one of the most commonly used in modelling heavily tailed network traffic. The Pareto type 2 distribution has 2 parameters (the tail shape parameter  $\alpha$  and the scale parameter  $x_m$ ). We use the Lomax distribution that is a shifted version of Pareto type 2 that allows generating values lower than the scale parameter, since it is used to model message interarrival times. We experimented by varying the value of the tail shape parameter,  $\alpha$ , to investigate the effect of changing the heaviness of the primary traffic tail on friendliness. The mean message interarrival time is also set to 100 ms and the scale parameter,  $x_m$ , of the Pareto distribution is calculated based on the value of  $\alpha$

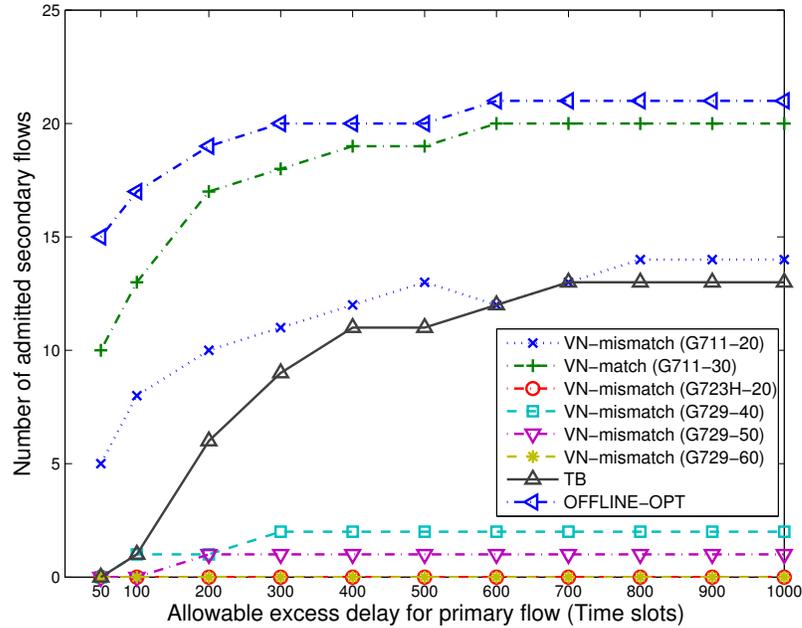


Figure 3.12: Secondary VoIP capacity vs. Primary Delay Tolerance for the VN Scheduler. Different virtual VoIP codec configurations are considered (codec mismatch case (G711-30 ms as the real codec configuration)).

and the required value of the distribution mean. For VoIP capacity calculation, the codec configuration used for the secondary VoIP call is G711 with a 30 ms packetization period. As discussed in [107],  $\alpha$  must be set greater than 1, since the Pareto distribution has both infinite mean and variance when  $\alpha$  is less than 1, which is not relevant to real packet traffic. Also, the distribution has finite mean and infinite variance when  $\alpha$  is less than 2. Hence, when  $\alpha$  is assigned values greater than 2, the distribution has both finite mean and variance. However, studies (e.g. [108] and [109]) have shown that values of  $\alpha$  between 1 and 2 best model real packet traffic. Therefore, in our simulations,  $\alpha$  is set between 1 and 3 and we added the range from 2 to 3 to investigate the effect of having finite variance on the performance of the algorithms.

Figures 3.13, 3.14 and 3.15 show the secondary VoIP capacity achieved by the TB

scheduler, VN scheduler in the perfect codec match and codec mismatch cases, respectively, for different values of the tail shape parameter,  $\alpha$ . It is shown that the smaller the value of  $\alpha$ , the heavier the tail of the distribution, which causes the achievable secondary VoIP capacity to decrease for a constant value of primary delay tolerance. It can also be seen that when  $\alpha$  is between 2.5 and 3, the schedulers' performance is very similar to that in the case of the Poisson arrival process presented in the previous sections (see Figure 3.12). This is due the fact that for these values of  $\alpha$ , the Pareto distribution has both finite mean and variance and a less heavier tail shape than for other values of  $\alpha$ . By comparing Figures 3.13, 3.14 and 3.15, it can be seen that the TB algorithm achieves a zero or very low secondary VoIP capacity for lower values of  $\alpha$ , while the VN scheduler (both perfect codec match and mismatch cases) shows better performance for the same parameters. However, as previously discussed, since the performance of the VN scheduler in the perfect codec match case is not achievable unless the secondary node has an adaptive VoIP codec transmission module, we compare with the performance of the VN scheduler in the codec mismatch case with G711-20 ms for the virtual VoIP calls, which is presented in Figure 3.15. It is also apparent that some of the curves in Figures 3.13 and 3.15 lose their monotonic property for some values of primary traffic delay tolerance, which is related to the heaviness of the tail of the distribution and the infinite variance of the Pareto distribution for values of  $\alpha$  less than 2.

### 3.6.6 Secondary Traffic QoS Requirements

In the results presented above, the performance of OFFLINE-OPT and the online friendly schedulers was studied under varying primary delay tolerance. In this section the performance of the online friendly schedulers is considered by varying the jitter requirement of

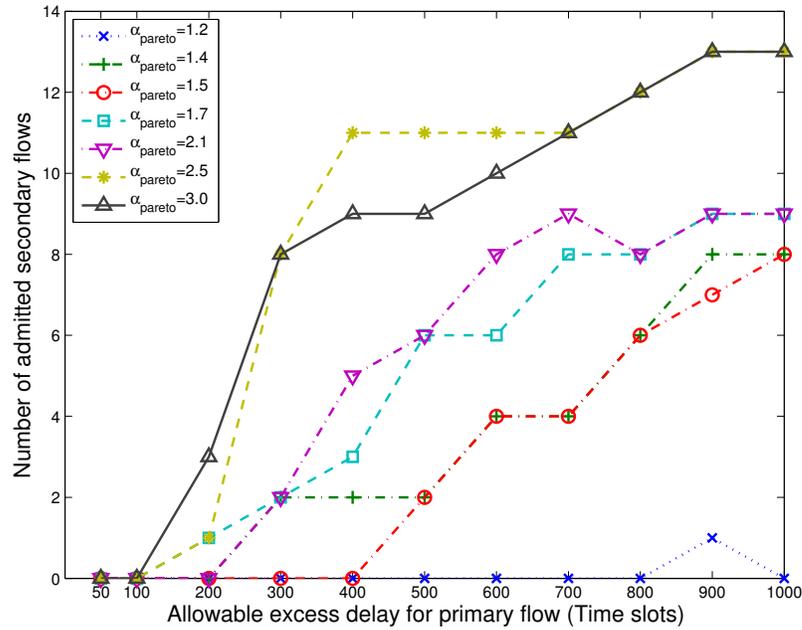


Figure 3.13: Secondary VoIP Capacity vs. Primary Delay Tolerance for the TB Scheduler. Different values of the Pareto tail shape parameter are considered (G711-30 ms as the codec configuration).

the secondary VoIP calls.

We assume that primary messages arrive to the basestation according to a Poisson arrival process with a mean inter-arrival time of 100 ms (i.e., mean message arrival rate = 1/100 messages/ms), and since the message size is 30 packets, the mean packet arrival rate is on average 1/3 packets/ms. The secondary VoIP call delay requirement is set to 100 ms [105]. The secondary VoIP call codec configuration used is G711 with 20 ms and 30 ms packetization periods. The value of the maximum allowable excess delay for the primary is set to 300 ms, as we are interested in delay tolerant primary traffic. The jitter requirement of the secondary VoIP calls varies from tight jitter requirements (i.e., 10 ms) to no jitter requirement at all, i.e., the jitter requirement is set to the delay requirement.

As expected, Figure 3.16 shows higher secondary VoIP capacity for relaxed secondary

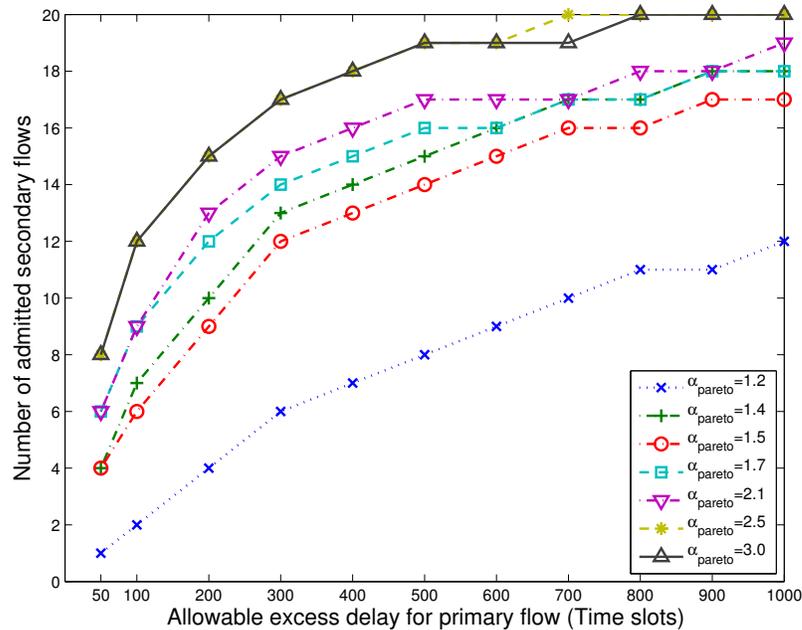


Figure 3.14: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VN Scheduler. Different values of the Pareto tail shape parameter are considered (perfect codec match case (G711-30 ms as the codec configuration)).

VoIP jitter requirements for a given scheduler and a given VoIP codec configuration. This is due to the fact that as the VoIP QoS requirements are relaxed, a higher number of VoIP calls can be supported. Relaxing the jitter requirement of a secondary VoIP packet increases the number of conforming time slots in which this packet can depart. It is also clear that the VN scheduler (perfect match case) still performs better than the TB scheduler. Also, as seen in the previous sections, secondary VoIP capacity for G711 with 30 ms is higher than that for G711-20 ms. This is due to the fact that both of these configurations transmit one secondary packet per single primary slot, and since G711-30 ms uses a smaller number of VoIP packets than G711-20 ms for a given VoIP call, this is translated into higher secondary VoIP capacity.

We have also experimented with varying the delay requirement of the secondary VoIP

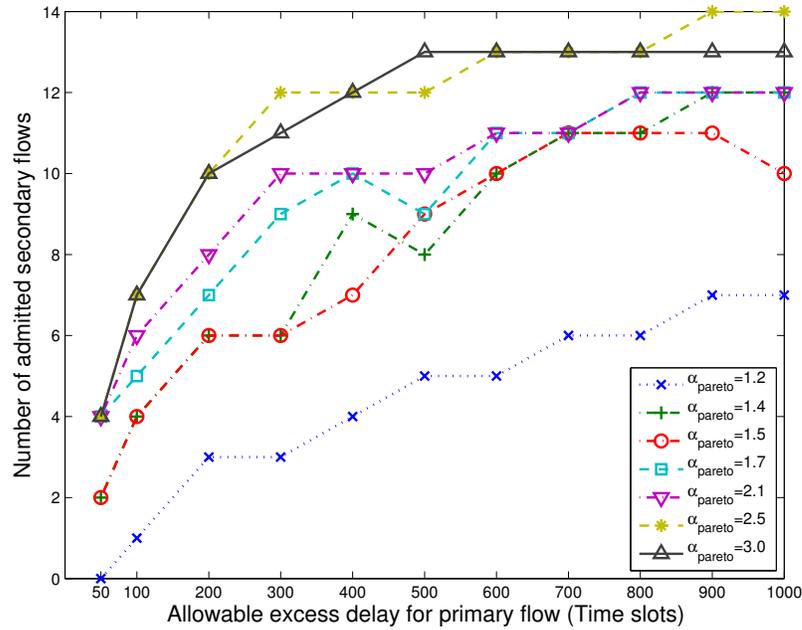


Figure 3.15: Secondary VoIP capacity vs. Primary Delay Tolerance for the VN Scheduler. Different values of the Pareto tail shape parameter are considered (codec mismatch case (G711-20 ms as the virtual codec configuration and G711-30 ms as the real codec configuration)).

calls for a given VoIP jitter requirement and a given primary delay tolerance. It was concluded that for a given jitter requirement, varying the delay requirement does not have a significant effect on the achievable secondary VoIP capacity. For this reason, this set of results was not included. It is also clear that for a given VoIP delay requirement, the VoIP jitter requirement controls the maximum number of secondary VoIP calls that can be supported. This explains why any friendly scheduler always reaches a saturation point, where increasing the primary delay tolerance does not further increase the achievable secondary VoIP capacity.

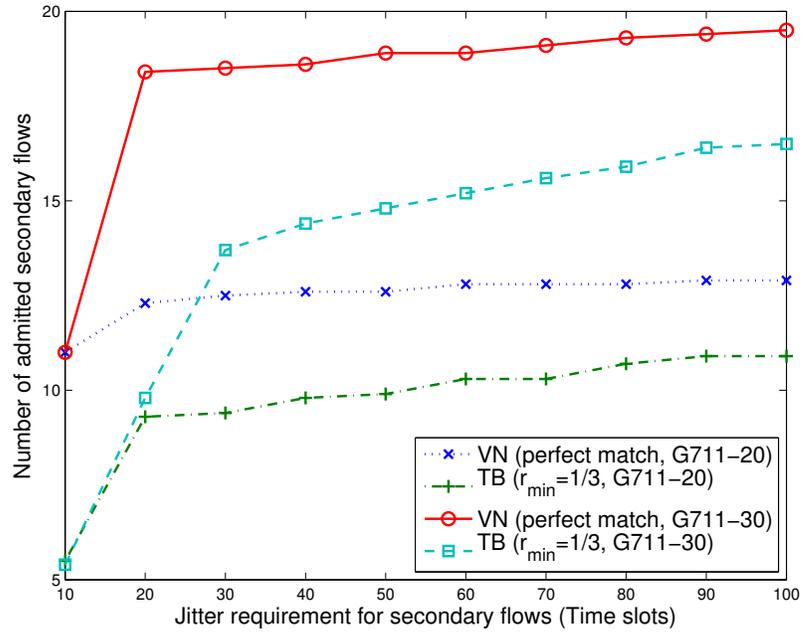


Figure 3.16: Secondary VoIP capacity vs. Secondary VoIP Call Jitter Requirement for the TB and VN Schedulers (perfect VN match case). G711 is the VoIP codec used with 20 ms and 30 ms packetization periods.

### 3.7 Conclusions

In this chapter we have proposed basestation scheduling mechanisms that are designed to be friendly from a secondary network user perspective. The chapter considered the case where the primary user transmits data packets subject to delay constraints and the secondary users are transmitting real-time VoIP traffic. First, a measure for the secondary user friendliness was defined that represents the secondary voice capacity that can be obtained. An offline scheduler was then formulated that maximizes this friendliness. This schedule can be found using a minimum cost flow graph construction in time complexity that is polynomial in the number of time slots. Two traffic schedulers were then proposed. The first (TB scheduler) is based on shaping the primary traffic, which shapes the spectrum holes available to secondary traffic, and the second (VN scheduler) is based on scheduling both

the real primary traffic and virtually modelled secondary VoIP traffic with priority given to the primary traffic.

The results showed that for both the proposed schedulers and the offline scheduling bound, the higher the delay tolerance of the primary traffic, the more secondary friendly it can be. Our results also show that the proposed schedulers perform significantly better than a conventional FCFS scheduler especially when primary traffic consists of bursty message based arrivals with a high delay tolerance. As expected, both of the proposed schedulers achieve lower performance than the non-causal friendliness upper bound.

In the TB scheduler, an adaptive token rate setting scheme is used to adjust the spectral hole shaping based on the current state of the primary traffic backlog that tends to create spectrum holes as friendly as possible. A minimum token rate threshold is set to avoid future primary QoS violation based on past scheduling decisions. Results showed that the average primary packet arrival rate can be considered a good candidate value for this threshold. For the VN scheduler, the cases were considered where there is a mismatch between the codec configuration of the virtual VoIP call and the real VoIP calls.

It was shown that the performance of the VN algorithm largely depends on the choice of the codec configuration of the virtual and real VoIP traffic. In the case of perfect codec match, the VN scheduler performs better than the TB scheduler. In the case of codec mismatch, the VN scheduler may achieve performance that is better or worse than the TB scheduler based on the codec configuration used. On the other hand, the TB scheduler performance is only dependent on the primary traffic, however it is completely non-friendly (zero secondary VoIP capacity) under very tight primary delay constraints when compared to the VN scheduler.

As for the primary message arrival process, the Poisson process has been used in most

of the simulations. However, we included results using the Pareto type 2 distribution to study the effect of heavy tailed primary traffic. Our results showed that the heavier the tail of the distribution, the lower the secondary VoIP capacity that can be achieved by the proposed schedulers. For less heavier tail shape configurations, the schedulers achieved performance very similar to that in the case of Poisson arrivals. Simulation results also showed that varying the QoS requirements of the secondary VoIP traffic, directly affect the achievable secondary VoIP capacity. In particular, the jitter requirement of the secondary VoIP call significantly affect the friendliness level of the primary network.

# **Chapter 4**

## **Secondary VoIP Capacity in Opportunistic Spectrum Access Networks with Friendly Scheduling (Variable Transmission Rate)**

### **4.1 Introduction**

In this chapter we focus on secondary VoIP friendliness and primary traffic schedule generation in the case of multiple primary data flows with variable bit rate (VBR) transmission, rather than the constant bit rate transmission case considered in the previous chapter. Similar to Chapter 3, we propose an offline and online approach to generate a primary traffic schedule that is friendly to potential secondary VoIP calls using the same friendliness measure. We first develop an offline VBR scheduler that can maximize friendliness over finite

time intervals. Then we present two online friendly VBR schedulers, VBR-TB and VBR-VN, which are modified versions of the previously proposed online friendly schedulers, TB and VN, to support multiple primary data flows with variable transmission bit rates. The modifications include aggregating packets from multiple primary data flows and consider variable transmission bit rates. Simulation results were presented that show the effect of the primary traffic load and primary network delay tolerance on the friendliness level of the primary network.

The remainder of this chapter is organized as follows. In Section 4.2 we give a detailed description of our system model. Section 4.3 then formulates an optimum offline primary schedule. In Section 4.4, the modified versions of our online friendly scheduling algorithms are presented, VBR-TB and VBR-VN, based on spectral hole shaping and virtual node scheduling, respectively. Section 4.5 presents the different levels of achievable secondary user friendliness under varying primary traffic load and primary traffic delay requirements. Finally, Section 4.6 presents the conclusions of this chapter.

## 4.2 System Model

Two collocated infrastructure based primary and secondary networks are considered that share a single licensed channel owned by the primary network, based on an *interweave* spectrum sharing model [10]. A time-invariant TDMA channel model is assumed, where multiple primary users receive multi-packet message data from the primary basestation at variable transmission bit rates. The number of time slots needed to carry a given primary packet depends on the slot duration, the primary packet size and the transmission bit rate of primary flow to which this packet belongs. We assume the same secondary network model and packet deadline assignment as in Section 3.3. Our goal is to design a secondary VoIP

user friendly and QoS-aware VBR scheduler for the primary basestation. The inputs and outputs of our VBR scheduling problem are more formally defined as follows.

**INPUT:** Packet scheduling is to occur over a set of contiguous time slots, denoted by  $\mathcal{T}$  while  $\sigma$  represents the duration of a single time slot. We define a set of  $N_p$  primary data flows given by  $\mathcal{F}_p$ . For each  $j \in \mathcal{F}_p$  we define  $\mathcal{K}_p(j)$ , to be the set of primary packets belonging to primary flow  $j$ . Then we define the  $\mathcal{U}_p$  to be the set of aggregated packets belonging to different primary flows

$$\mathcal{U}_p = \cup_{j \in \mathcal{F}_p} \mathcal{K}_p(j) \quad (4.1)$$

Every primary packet,  $i \in \mathcal{U}_p$ , has a packet length in bits  $L_p(i)$ , an arrival time and a delay constraint, and we define

$$\mathcal{T}_p(i) = \{ t \in \mathcal{T} \mid \text{time slot } t \text{ satisfies the QoS constraints for primary packet } i. \} \quad (4.2)$$

i.e.,  $\mathcal{T}_p(i) \subset \mathcal{T}$  is the set of time slots, any one of which would satisfy packet  $i$ 's QoS constraints if assigned to that packet. Since the primary traffic has some delay tolerance, for every primary data flow,  $j \in \mathcal{F}_p$ , it is allowed that a fraction of packets,  $\epsilon$ , can be dropped with no scheduling when their delay constraints were violated. Clearly, an  $\epsilon$  of 1 corresponds to completely unconstrained delay tolerance and a value of 0 corresponds to requiring all primary delay constraints be met. Since every primary flow may use a different transmission bit rate, the scheduler is given the number of bits that can be carried in each time slot  $t \in \mathcal{T}$  for primary packet  $i$ , denoted by  $b(i)$ , which depends on the channel model. Obviously, the value of  $b(i)$  is the same for all packets  $i \in \mathcal{K}_p(j)$  belonging to the same primary flow  $j \in \mathcal{F}_p$ .

In a similar way, we define a candidate set of  $F_s$  secondary VoIP flows given by  $\mathcal{F}_s$ . For each  $n \in \mathcal{F}_s$  we define  $\mathcal{K}_s(n)$ , to be the set of secondary packets belonging to secondary flow  $n$ . Then we define  $\mathcal{U}_s$  to be the set of aggregated packets belonging to different secondary VoIP flows.

$$\mathcal{U}_s = \cup_{n \in \mathcal{F}_s} \mathcal{K}_s(n) \quad (4.3)$$

Every secondary packet  $m \in \mathcal{U}_s$  has an arrival time and delay/jitter requirement, and we define

$$\mathcal{T}_s(m) = \{ t \in \mathcal{T} \mid \text{time slot } t \text{ satisfies the QoS constraints for secondary packet } m. \} \quad (4.4)$$

i.e.,  $\mathcal{T}_s(m) \subset \mathcal{T}$  is the set of conforming time slots for secondary VoIP packet  $m$ . Also, we define  $\beta$  to be the fraction of packets in each VoIP flow,  $n \in \mathcal{F}_s$ , which can be dropped with no scheduling. Unlike the primary network case, the number of secondary VoIP packets that can be carried in any given time slot is fixed for all secondary VoIP flows and is denoted as  $n_s$ .

**OUTPUT:** The objective is for the primary network basestation to schedule packet transmissions aggregated from different  $N_p$  primary flows with variable transmission bit rate subject to meeting both its own primary packets deadline constraints and the QoS constraints for the  $F_s$  secondary VoIP flows. More formally, we define

$$x_p(i, t) = \begin{cases} 1 & \text{if primary packet } i \text{ is assigned to time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (4.5)$$

for all  $i \in \mathcal{U}_p$  and  $t \in \mathcal{T}$ .

$$d_p(i) = \begin{cases} 1 & \text{if primary packet } i \text{ is dropped with no scheduling,} \\ 0 & \text{otherwise.} \end{cases} \quad (4.6)$$

for all  $i \in \mathcal{U}_p$ . Similarly we define

$$x_s(m, t) = \begin{cases} 1 & \text{if secondary packet } m \text{ is assigned to time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (4.7)$$

for all  $m \in \mathcal{U}_s$  and  $t \in \mathcal{T}$ .

$$y_s(t) = \begin{cases} 1 & \text{if there exists at least one secondary packet } m \text{ for which } x_s(m, t) = 1, \\ 0 & \text{otherwise.} \end{cases} \quad (4.8)$$

for all  $t \in \mathcal{T}$ .

$$d_s(m) = \begin{cases} 1 & \text{if secondary packet } m \text{ is dropped with no scheduling,} \\ 0 & \text{otherwise.} \end{cases} \quad (4.9)$$

for all  $m \in \mathcal{U}_s$ . Given these definitions and the above inputs, the objective for the primary network is to assign values for  $x_p(i, t)$  and  $d_p(i)$  that satisfies its own packet deadline constraints given above. A friendly scheduler will also try to do this in a way that best accommodates the secondary network traffic.

As in Chapter 3, a friendly scheduler can generate either *offline* or *online* schedules. A VBR *offline scheduler* is provided with the complete set of input sample functions for

both the primary and secondary packets (i.e., for a given set of  $F_s$  VoIP flows), which it then uses to generate an offline schedule. The information known a priori includes all primary packet lengths,  $L_p(i)$ , all primary and secondary packet time slot conforming sets  $\mathcal{T}_p(i)$  and  $\mathcal{T}_s(m)$ , for all  $i \in \mathcal{U}_p$  and  $m \in \mathcal{U}_s$ . It also includes transmission bit rates for all primary flows  $b(i)$ , for all  $i \in \mathcal{U}_p$  as well as secondary VoIP codec type and configuration. An optimal offline schedule can be used to compute an upper bound on friendliness, as shown in Section 4.3. In contrast to offline scheduling, a VBR *online scheduler* is provided with the primary packet inputs in real time and it must make causal downlink time slot assignments. In this case, neither  $\mathcal{F}_s$  nor any of the VoIP packet arrival details are known to the basestation. In Section 4.4, VBR online scheduling algorithms are discussed. Before proposing the VBR online schedulers, we first derive an optimal offline VBR scheduler that maximizes the number of secondary VoIP flows that can be scheduled for given sets of input sample functions. This is introduced in the next section and is used for comparisons with the VBR online schedulers later.

## 4.3 Optimum Offline VBR Scheduling and Friendliness

### Upper Bound

The optimization expressed as a feasibility problem is presented below and is solved for a given set of  $F_s$  secondary VoIP calls, i.e., assignments are made for  $x_s(m, t)$ ,  $y_s(t)$ ,  $x_p(i, t)$ ,  $d_p(i)$  and  $d_s(m)$ . The bound is obtained by increasing the number of flows in  $\mathcal{F}_s$  until an infeasible solution is reached. In other words, the optimization problem formulated below is solved for a predetermined set of secondary flows,  $\mathcal{F}_s$  and the primary traffic, subject to their QoS constraints. If a feasible solution is found, then the primary network can

accommodate that level of friendliness. When this is the case, a new VoIP flow is added and the test is repeated. Eventually, the maximum number of VoIP flows is determined once the optimization reaches infeasibility. Using the definitions from Section 4.2, the feasibility test, denoted by OFFLINE-VBR-OPT, is as follows.

$$\text{minimize } 0 \quad (\text{OFFLINE-VBR-OPT})$$

$$\text{subject to } \sum_{i \in \mathcal{U}_p} x_p(i, t) + y_s(t) \leq 1, \quad \forall t \in \mathcal{T} \quad (4.10)$$

$$\sum_{m \in \mathcal{U}_s} x_s(m, t) \leq n_s, \quad \forall t \in \mathcal{T} \quad (4.11)$$

$$y_s(t) \geq x_s(m, t), \quad \forall m \in \mathcal{U}_s, t \in \mathcal{T} \quad (4.12)$$

$$y_s(t) \leq \sum_{m \in \mathcal{U}_s} x_s(m, t), \quad \forall t \in \mathcal{T} \quad (4.13)$$

$$\sum_{t \in \mathcal{T}_p(i)} b(i)x_p(i, t) \geq (1 - d_p(i))L_p(i), \quad \forall i \in \mathcal{U}_p \quad (4.14)$$

$$\sum_{t \in \mathcal{T}_s(m)} x_s(m, t) \geq (1 - d_s(m)), \quad \forall m \in \mathcal{U}_s \quad (4.15)$$

$$\sum_{i \in \mathcal{K}_p(j)} d_p(i) \leq \epsilon |\mathcal{K}_p(j)|, \quad \forall j \in \mathcal{F}_p \quad (4.16)$$

$$\sum_{m \in \mathcal{K}_s(n)} d_s(m) \leq \beta |\mathcal{K}_s(n)|, \quad \forall n \in \mathcal{F}_s \quad (4.17)$$

$$x_p(i, t) \in \{0, 1\}, \quad \forall i \in \mathcal{U}_p, t \in \mathcal{T} \quad (4.18)$$

$$x_s(m, t) \in \{0, 1\}, \quad \forall m \in \mathcal{U}_s, t \in \mathcal{T} \quad (4.19)$$

$$y_s(t) \in \{0, 1\}, \quad \forall t \in \mathcal{T} \quad (4.20)$$

$$d_p(i) \in \{0, 1\}, \quad \forall i \in \mathcal{U}_p \quad (4.21)$$

$$d_s(m) \in \{0, 1\}, \quad \forall m \in \mathcal{U}_s \quad (4.22)$$

The constant objective in OFFLINE-VBR-OPT constructs a feasibility test for the input data and the following constraints. The constraints in OFFLINE-VBR-OPT are based on the assumptions mentioned in Section 4.2. Constraint (4.10) enforces the single channel assumption, where a single time slot can be used for either primary transmission or secondary transmission. Constraint (4.10) ensures that in the case of primary transmission, a given time slot  $t \in \mathcal{T}$  is assigned to a single primary packet. Constraint (4.11) ensures that only  $n_s$  secondary packets can depart at the same time slot  $t \in \mathcal{T}$ . Constraints (4.12) and (4.13) establish the relation between the decision variables  $x_s(m, t)$  and  $y_s(t)$  for all  $m \in \mathcal{U}_s$  and  $t \in \mathcal{T}$ , to make sure that  $y_s(t) = 1$  only if time slot  $t$  is scheduled for secondary transmission and 0 otherwise. Constraint (4.14) ensures that a given scheduled, non-dropped, primary packet should be assigned the number of time slots that fulfill that packet length requirement considering different transmission rates of different primary flows. Similarly, constraint (4.15) guarantees that a given scheduled, non-dropped, secondary packet is assigned a time slot. Constraint (4.16) preserves the QoS constraints of the primary packets, by ensuring that no more than the maximum allowable drop fraction  $\epsilon$  do not meet their QoS requirements for every primary flow  $j \in \mathcal{F}_p$ . Similarly, constraint (4.17) ensures the QoS per flow guarantees for secondary packets. Constraints (4.18) to (4.22) define  $x_s(m, t)$ ,  $x_p(i, t)$ ,  $y_s(t)$ ,  $d_p(i)$  and  $d_s(m)$  as binary decision variables.

The ILP formulated above, OFFLINE-VBR-OPT, can be relaxed to a less complex conventional linear program by removing the binary constraints of the decision variables. However, the formulated ILP is solvable for short simulation time intervals as shown later in Section 4.5.

## 4.4 Online Secondary Friendly VBR Scheduling

In this section we propose two online secondary friendly VBR schedulers. The algorithms operate at the primary basestation and make scheduling decisions at each time slot based on the pending primary traffic backlog from multiple primary flows and its QoS requirements. The algorithms are based on the online friendly schedulers proposed in Section 3.5 and modified to consider multiple primary flows with variable transmission bit rates.

Similar to the TB scheduler proposed in Section 3.5.1, the first algorithm, *Adaptive Token Bucket Spectrum Hole Shaping VBR Scheduler (VBR-TB)*, achieves secondary VoIP friendliness by attempting to create periodically spaced spectrum holes for potential secondary voice transmissions. The VBR-TB scheduler shapes the primary traffic, aggregated from multiple primary flows, to prevent primary bursts (messages) from occupying the channel for periods of time that would be inconsistent with secondary QoS requirements. For this purpose, we use a simple token bucket where the token rate is dynamically set when new primary messages arrive, based on the QoS requirements and the variable transmission rates of different primary flows.

Similar to the VN scheduler proposed in Section 3.5.2, the second algorithm, *Virtual Node Scheduler (VN)*, uses a real-time scheduler to coordinate both *real* primary traffic and *virtual* secondary VoIP flows. Ties in contention are resolved in favor of the primary traffic. The performance of the virtual node scheduler depends on the differences between the assumed virtual VoIP codec configuration and the real VoIP codec configuration. Both perfect codec match and different cases of codec mismatch between the real and virtual codecs are considered. Unlike the VN scheduler, the VBR-VN scheduler considers multiple primary flows with variable transmission rates when making scheduling decisions. Details of the two VBR schedulers are introduced in the following two sections.

#### 4.4.1 Adaptive Token Bucket Spectrum Hole Shaping VBR Scheduler (VBR-TB)

The VBR-TB algorithm takes the same approach as the TB scheduler, discussed in Section 3.5.1. At any given time slot, the VBR-TB scheduler aggregates message based data traffic from multiple primary data flows, then shapes it using a token based approach based on their deadline requirements and variable transmission rates, in a way that creates VoIP compatible spectral hole patterns. The adaptive token bucket algorithm parameters,  $r$  and  $W$ , are defined and set as in Section 3.5.1. And the minimum token rate threshold,  $r_{min}$ , is set to the primary packet mean arrival rate averaged over all primary flows, taking into consideration different transmission rates of different primary flows.

A more formal description of the scheduler is shown in Algorithm 1. At any given time slot  $t$ , in Steps 2 to 7, the VBR-TB scheduler checks for new primary packet arrivals and execute specific steps upon new arrivals. In Step 3 the current pending primary traffic backlog  $\mathcal{U}_p(t)$  and the most urgent primary packet index  $e(t) \in \mathcal{U}_p(t)$  are updated to reflect online packet arrivals at time slot  $t$ . Step 4 initializes  $N(i, t)$ , to the number of time slots needed to transmit a given primary packet  $i$ , which has newly arrived at the current time slot  $t$ . This is computed based on the packet size  $L_p(i)$ , the slot duration  $\sigma$  and the packet transmission bit rate  $b(i)$ . Different transmission bit rates of different primary users are considered, since the VBR-TB scheduler keeps track of the number of time slots assigned to every pending primary packet at every time slot, until a packet is fully transmitted. Step 5 defines the current new scheduling window time frame  $\mathcal{T}_{SW}$ , which starts at the current time slot  $t$  and ends at the farthest known future deadline  $t_d$ . Moreover, the current token rate value,  $r(t)$ , is recomputed every time a new primary message arrives. This is shown in Step 6 where the current value of the token rate,  $r(t)$ , is updated to the minimum possible

**Algorithm 1** VBR-TB Algorithm

$\mathcal{T}$  : set of all time slots  
 $\mathcal{T}_{SW}$  : set of time slots in the current scheduling window, redefined upon new message arrivals,  
 $\mathcal{T}_{SW} \subset \mathcal{T}$   
 $\mathcal{U}_p(t)$  : aggregated primary packet backlog from all  $\mathcal{F}_p$  flows at time slot  $t$   
 $e(t)$  : index of the primary packet with the earliest deadline among  $\mathcal{U}_p(t)$  at time slot  $t$   
 $\mathcal{R}$  : set of possible values of token generation rate, i.e.,  $\mathcal{R} = \{1, \frac{1}{2}, \frac{1}{3}, \dots, r_{min}\}$   
 $r(t)$  : current value of token generation rate at time slot  $t$   
 $B(t)$  : binary current bucket level at time slot  $t$ , i.e.,  $B(t) = 1$  if a token exists, 0 otherwise  
 $L_p(i)$  : packet length in bits of primary packet  $i$ ,  $\forall i \in \mathcal{U}_p(t)$   
 $\sigma$  : duration of a single time slot in seconds  
 $b(i)$  : transmission bit rate of primary packet  $i$ ,  $\forall i \in \mathcal{U}_p(t)$   
 $N(i, t)$  : number of time slots needed to complete primary packet  $i$  transmission at time slot  $t$ ,  $\forall i \in \mathcal{U}_p(t), \forall t \in \mathcal{T}$   
 $A(i, t)$  : number of available conforming time slots for primary packet  $i$  at time slot  $t$ ,  $\forall i \in \mathcal{U}_p(t), \forall t \in \mathcal{T}$

```

1: for all  $t \in \mathcal{T}$  do
2:   if a new primary message arrives then
3:     Update  $\mathcal{U}_p(t)$  and  $e(t)$ .
4:     Initialize  $N(i, t) = \left\lceil \frac{L_p(i)}{\sigma b(i)} \right\rceil$ ,  $\forall i \in$  set of new primary packets arriving at time slot  $t$ .
5:     Define  $\mathcal{T}_{SW} = \{t, t + 1, \dots, t_d\}$ , where  $t_d$  is the largest primary packet deadline.
6:     Set  $r(t)$  to the minimum token rate value  $\in \mathcal{R}$ , that satisfies the QoS of  $\mathcal{U}_p(t)$  packets over  $\mathcal{T}_{SW}$ .
7:   end if
8:   TB-GENERATE-TOKEN( $t, r(t), B(t)$ )
9:   while pending primary packets exist and a token is ready, i.e.,  $|\mathcal{U}_p(t)| > 0$  &  $B(t) = 1$  do
10:    Set  $A(e(t), t) = \lfloor r(t)(\text{Deadline of primary packet } e(t) - t) \rfloor + 1$ .
11:    if  $N(e(t), t) \leq A(e(t), t)$  then
12:      Send a fragment of primary packet  $e(t)$  at time slot  $t$ .
13:      Update  $N(e(t), t) = N(e(t), t) - 1$ .
14:      if  $N(e(t), t) = 0$  then primary packet  $e(t)$  transmission complete. end if
15:      Update current bucket level  $B(t)$ , i.e.,  $B(t) = 0$ .
16:    else
17:      Drop primary packet  $e(t) \in \mathcal{U}_p(t)$ .
18:    end if
19:    Update  $\mathcal{U}_p(t)$  and  $e(t)$ .
20:  end while
21: end for
22: function TB-GENERATE-TOKEN( $t, r(t), B(t)$ )
23:   if bucket is empty and it is time for a new token, i.e.,  $B(t) = 0$  &  $t \bmod \frac{1}{r(t)} = 0$  then
24:     Generate a new token, i.e.,  $B(t) = 1$ 
25:   end if
26:   return  $B(t)$ 
27: end function

```

value greater than  $r_{min}$  that can satisfy the delay requirements for the current pending primary packet backlog  $\mathcal{U}_p(t)$ , as of time slot  $t$ .

In Step 8, the TB-GENERATE-TOKEN function call is responsible for new token generation. The TB-GENERATE-TOKEN function is declared in Steps 22 to 27 and accepts the current token rate  $r(t)$ , the current time slot  $t$  and the current bucket level  $B(t)$ . It then returns an updated bucket level that reflects whether a new token has been generated. In Steps 9 to 20, the current token rate value  $r(t)$  is used to shape the current primary backlog. As long as a token is available and at least one pending primary packet exists, the VBR-TB keeps searching for the most urgent primary packet,  $e(t)$ , which can be fully transmitted before its deadline. Step 10 computes  $A(e(t), t)$ , which is the number of available conforming time slots for the most urgent primary packet  $e(t)$  taking into consideration the current token rate  $r(t)$ . In Step 11,  $A(e(t), t)$  is compared against the current number of time slots needed to finish transmitting primary packet  $e(t)$  as of time slot  $t$ , which is denoted by  $N(e(t), t)$ . If  $N(e(t), t) \leq A(e(t), t)$ , Step 12 sends a fragment of primary packet  $e(t)$  at time slot  $t$  and  $N(e(t), t)$  is decremented by one in Step 13. If  $N(e(t), t) = 0$ , this means  $e(t)$  is completely transmitted and it is marked to be removed from the current pending primary packet backlog  $\mathcal{U}_p(t)$ , this check is done in Step 14. In Step 15, the current bucket level  $B(t)$ , is updated to reflect that primary traffic has occupied time slot  $t$ . On the other hand, in Step 17, if  $N(e(t), t) > A(e(t), t)$  the current most urgent packet  $e(t)$  is dropped, and  $e(t)$  is set to the next most urgent packet in the current primary backlog  $\mathcal{U}_p(t)$ . Essentially,  $\mathcal{U}_p(t)$  and  $e(t)$  are updated on a per time slot basis to reflect online departures and droppings of primary packets, as shown in Step 19. Hence, the while loop in Steps 9 to 20 ensures that the VBR-TB scheduler does not assign any given time slot  $t$  to a primary packet that will have to be dropped later due to lack of availability of conforming time slots

that satisfy the packet's deadline, taking into consideration the current token rate  $r(t)$ , used for primary traffic shaping. This ensures efficient primary channel utilization while satisfying the primary traffic constraints, in a way that is friendly to periodic secondary VoIP calls.

Some implementation details are not shown in the algorithm description for the sake of clarity. Specifically, at every time slot  $t$ , each of  $B(t)$  and  $r(t)$  are initialized to the bucket level and the token rate set at the previous time slot  $t-1$ , respectively, i.e.,  $B(t) \leftarrow B(t-1)$  and  $r(t) \leftarrow r(t-1)$ . At every time slot  $t$ ,  $N(i, t)$  is implicitly initialized to the number of time slots needed to fully transmit every pending primary packet as of time slot  $t-1$ , i.e.,  $N(i, t) \leftarrow N(i, t-1)$  for all  $i \in \mathcal{U}_p(t)$ .

#### 4.4.2 Virtual Node VBR Scheduler (VBR-VN)

The VBR-VN algorithm takes the same approach as the VN scheduler, discussed in Section 3.5.2. At any given time slot, the VBR-VN scheduler aggregates message based data traffic from real multiple primary data flows, then schedules it for transmission along with virtual secondary VoIP traffic flows, under packet deadline constraints. The virtual secondary voice capacity,  $F_{vs}^*$ , of the VBR-VN scheduler is defined and computed as in Section 3.5.2.

A more formal description of the algorithm is shown in Algorithm 2. At time  $t$  it deals with two types of queues, the real primary traffic queue,  $Q_p(t)$ , which carries aggregated traffic from multiple primary data flows, defined as  $\mathcal{F}_p$  and the virtual secondary traffic queue,  $Q_s(t)$ , which carries aggregated traffic from the virtual secondary VoIP flows that belong to the set of virtual secondary flows, defined as  $\mathcal{F}_{vs}$ . Both queues are sorted with the packet having the earliest deadline at the head of the queue, denoted by  $HoQ_p(t)$  and  $HoQ_s(t)$  for the primary and secondary queues, respectively. Similar to the VBR-TB

**Algorithm 2** VBR-VN Algorithm

$\mathcal{T}$  : set of all time slots  
 $\mathcal{F}_{vs}$  : set of virtual secondary flows  
 $Q_p(t)$  : current aggregated queue of primary packets belonging to  $\mathcal{F}_p$  flows at time slot  $t$  sorted with earliest deadline first (EDF)  
 $Q_s(t)$  : current *virtual* aggregated queue of secondary packets belonging to  $\mathcal{F}_{vs}$  flows at time slot  $t$  sorted with earliest deadline first (EDF)  
 $HoQ_p(t)$ : index of head of queue packet of  $Q_p(t)$   
 $HoQ_s(t)$ : index of head of queue packet of  $Q_s(t)$   
 $n_s$  : number of secondary packets that fit in a single time slot, which is calculated based on the *virtual* codec type and configuration used  
 $L_p(i)$  : packet length in bits of primary packet  $i, \forall i \in Q_p(t)$   
 $\sigma$  : duration of a single time slot in seconds  
 $b(i)$  : transmission bit rate of primary packet  $i, \forall i \in Q_p(t)$   
 $N(i, t)$  : number of time slots needed to complete primary packet  $i$  transmission at time slot  $t, \forall i \in Q_p(t), \forall t \in \mathcal{T}$   
 $A(i, t)$  : number of available conforming time slots for primary packet  $i$  at time slot  $t, \forall i \in Q_p(t), \forall t \in \mathcal{T}$

```

1: for all  $t \in \mathcal{T}$  do
2:   if a new primary message arrive then
3:     Update  $Q_p(t)$  and  $HoQ_p(t)$ 
4:     Initialize  $N(i, t) = \left\lceil \frac{L_p(i)}{\sigma b(i)} \right\rceil, \forall i \in$  set of new primary packets arriving at time slot  $t$ .
5:   end if
6:   while pending packets exist, i.e.,  $|Q_p(t)| > 0$  or  $|Q_s(t)| > 0$  and time slot  $t$  is not assigned do
7:     if Deadline of primary packet  $HoQ_p(t) >$  Deadline of virtual secondary packet  $HoQ_s(t)$  then
8:       Assign time slot  $t$  to the first  $n_s$  secondary packets of virtual secondary queue  $Q_s(t)$ .
9:     else
10:      Set  $A(HoQ_p(t), t) = (\text{Deadline of primary packet } HoQ_p(t) - t) + 1$ .
11:      if  $N(HoQ_p(t), t) \leq A(HoQ_p(t), t)$  then
12:        Assign time slot  $t$  to primary packet  $HoQ_p(t)$ , send a fragment of  $HoQ_p(t)$ .
13:        Update  $N(HoQ_p(t), t) = N(HoQ_p(t), t) - 1$ .
14:        if  $N(HoQ_p(t), t) = 0$  then primary packet  $HoQ_p(t)$  transmission complete. end if
15:      else
16:        Drop primary packet  $HoQ_p(t)$ .
17:      end if
18:      Update  $Q_p(t)$  and  $HoQ_p(t)$ .
19:    end if
20:  end while
21: end for

```

scheduler, in Steps 2 to 5, the VBR-VN scheduler checks for new primary packet arrivals at time slot  $t$ . In Step 3 the current pending primary traffic backlog  $Q_p(t)$  and the most urgent primary packet index  $HoQ_p(t) \in Q_p(t)$  are updated to reflect online packet arrivals at time slot  $t$ . Then Step 4 initializes  $N(i, t)$  to the number of time slots needed to transmit a given primary packet  $i$ , which has newly arrived at current time slot  $t$ . This is computed using the same equation used in the VBR-TB scheduler, which considers different transmission rates of different primary flows.  $N(i, t)$  is decremented over the simulation time every time a fragment of packet  $i$  is assigned to a time slot, until packet  $i$  is fully transmitted.

At every time slot  $t$ , Steps 6 to 20 are repeated as long as at least one pending primary or secondary packet exists and time slot  $t$  is not yet assigned to real primary traffic or virtual secondary traffic. The VBR-VN scheduler keeps searching for the most urgent primary packet,  $HoQ_p(t)$ , which can be fully transmitted before its deadline or the most urgent secondary packet,  $HoQ_s(t)$ . Hence, Step 7 performs a comparison between the deadlines of  $HoQ_p(t)$  and  $HoQ_s(t)$ . In Step 8, if the head of queue packet of virtual secondary traffic,  $HoQ_s(t)$ , has the earliest deadline, the first  $n_s$  secondary packets of the virtual aggregated secondary queue,  $Q_s(t)$ , will be assigned to the current time slot,  $t$ . This declares time slot  $t$  as a spectral hole through virtual secondary packets assignments. Otherwise, similar to the VBR-TB scheduler, in Steps 10 to 18 the VBR-VN scheduler checks if the current most urgent  $HoQ_p(t)$  can be fully transmitted before its deadline or otherwise dropped and deadline comparisons are repeated between the updated  $HoQ_p(t)$  and  $HoQ_s(t)$ . Step 10 computes  $A(HoQ_p(t), t)$ , which is the number of available conforming time slots for the most urgent primary packet  $HoQ_p(t)$ . In Step 11,  $A(HoQ_p(t), t)$  is compared against the current number of time slots needed to finish transmitting primary packet  $HoQ_p(t)$  as of time slot  $t$ , which is denoted by  $N(HoQ_p(t), t)$ . If  $N(HoQ_p(t), t) \leq A(HoQ_p(t), t)$ ,

Step 12 assigns time slot  $t$  to primary packet  $HoQ_p(t)$  and sends a fragment of it at current time  $t$ . Then  $N(HoQ_p(t), t)$  is decremented by one in Step 13. If  $N(HoQ_p(t), t) = 0$ , this means  $HoQ_p(t)$  is completely transmitted and it is marked to be removed from the current pending primary packet backlog  $Q_p(t)$ , this check is done in Step 14. On the other hand, in Step 16, if  $N(HoQ_p(t), t) > A(HoQ_p(t), t)$  the current most urgent packet  $HoQ_p(t)$  is dropped, and  $HoQ_p(t)$  is set to the next most urgent packet in the current primary backlog  $Q_p(t)$ . Essentially,  $Q_p(t)$  and  $HoQ_p(t)$  are updated on a per time slot basis to reflect online departures and droppings of primary packets, as shown in Step 18. Hence, similar to the VBR-TB scheduler, the while loop in Steps 6 to 20 ensures that the VBR-VN scheduler does not assign any given time slot  $t$  to a primary packet that will have to be dropped later due to a lack of availability of conforming time slots that satisfy the packet's deadline. This ensures efficient primary channel utilization while satisfying the primary traffic constraints, in a way that is friendly to periodic secondary VoIP calls.

As discussed in Section 3.5.2, if the deadline of primary packet  $HoQ_p(t)$  equals the deadline of secondary packet  $HoQ_s(t)$ , the VBR-VN scheduler resolves the tie by assigning time slot  $t$  to the primary packet and dropping the virtual secondary packet, which prioritizes the real primary traffic. However, this is only done if the most urgent primary packet can be fully transmitted before its deadline, otherwise the most urgent primary packet is dropped and the deadline comparison is repeated as mentioned earlier. Essentially,  $Q_s(t)$  are updated every time slot based on online arrivals and departures; this detail is not shown in the algorithm steps. The virtual secondary traffic scheduler follows the secondary schedule assigned at step 8, which, in turn, updates the  $Q_s(t)$  queue. At every time slot  $t$ ,  $N(i, t)$  is implicitly initialized to the number of time slots needed to fully transmit every pending primary packet as of time slot  $t - 1$ , i.e.,  $N(i, t) \leftarrow N(i, t - 1)$  for all  $i \in Q_p(t)$ .

Similar to the online friendly schedulers proposed in Section 3.5, the VBR-TB and VBR-VN online schedulers require no prior knowledge of the primary traffic arrival process distribution. However as explained in Section 4.4.1, the VBR-TB algorithm tracks the mean arrival rate of each primary flow, to set a lower bound on the token bucket rate used to shape the primary traffic. The results presented in Section 4.5 show how the secondary friendliness level of a primary network is affected as the primary traffic load changes.

## 4.5 Simulation Model and Results

In this section we investigate the secondary voice capacity when a primary network, with VBR data transmissions, applies our proposed friendly scheduling at the primary basestation. Two main factors are considered: the primary network traffic load and the primary network delay tolerance. The primary network traffic load is determined by the per flow primary packet size and mean packet arrival rate. We assume the same secondary network model, primary network delay tolerance definition and secondary VoIP capacity computation approach as in Section 3.6. Under our proposed framework, the primary traffic is transmitted from the primary basestation to multiple primary users, using different transmission rates. This is done in a way that creates friendly spectral hole patterns for potential secondary VoIP calls. Therefore, due to secondary friendliness, the primary traffic suffers delay degradation within the primary network delay tolerance allowance, but no primary throughput degradation occurs.

Consider 3 primary users, each receiving a downlink transmission flow from the primary basestation at a different transmission bit rate. Assume primary Poisson process multi-packet message data arrivals for every primary transmission flow, with fixed message length and packet size. In order to reduce the number of system parameters, all primary

data flows have identical traffic descriptors, i.e., the values of the primary packet size, the primary message length and the Poisson process message mean arrival rate are the same for all flows. The time slot duration,  $\sigma$ , is taken to be 1 ms and the number of time slots needed to carry a given primary packet depends on the transmission data rate of the primary flow to which this packet belongs. We assume a simplified case where only three bit rate levels are used,  $H$  (high),  $M$  (medium) and  $L$  (low). For simulation purposes, we assume  $H = 12$  Mbps,  $M = 8$  Mbps and  $L = 4$  Mbps. Hence, for a fixed primary packet size of 1500 bytes for all primary flows, we assume that one  $H$  Mbps time slot is required to transmit one primary packet. We also assume that two and three  $M$  and  $L$  Mbps time slots are needed to accommodate one primary packet, respectively. The secondary VoIP packet size is calculated as in Section 3.6. The number of secondary packets that fit in a single 1 ms time slot depends on the VoIP codec configuration used. We also assume a fixed transmission bit rate for all secondary VoIP traffic, set to  $L$  Mbps. As for the secondary VoIP call QoS requirements, the delay and jitter requirements are set to 100 ms and 20 ms, respectively [105]. The secondary VoIP call codec configuration used is G711 with a 20 ms packetization period for OFFLINE-VBR-OPT and the online friendly VBR schedulers, VBR-TB and VBR-VN. The performance of the VBR-VN scheduler in the perfect codec match case and a representative case of codec mismatch is investigated. G711 with a 30 ms packetization period is used as the virtual VoIP codec configuration for the VBR-VN scheduler in the codec mismatch case. For G711 with 20ms or 30ms VoIP codec configurations, a single time slot of 1 ms duration can carry only one secondary packet at  $L$  Mbps.

For the simulation results presented below, the maximum packet loss ratio for primary traffic is set to zero, while the maximum packet loss ratio for a secondary VoIP call is set to

3% [59]. MATLAB was used for primary/secondary traffic generation and for implementing the online friendly VBR schedulers. The CVX convex optimization package [110] was used to model and solve the offline secondary friendly VBR scheduler OFFLINE-VBR-OPT.

The online VBR schedulers are first compared to the offline scheduling bound derived in Section 4.3 under varying primary traffic delay tolerance. Then results showing the performance of the VBR-VN and VBR-TB schedulers under different levels of primary traffic load are presented.

#### 4.5.1 Optimum Offline vs. Online VBR Schedulers

In this section we consider the performance of the proposed online VBR schedulers with varying primary packet delay tolerance compared to the optimum offline VBR scheduling bound derived in Section 4.3. Consider 3 primary multi-packet message data arrivals with identical traffic descriptors, each transmitted from the primary basestation to a primary user at a different transmission bit rate ( $H$  or  $M$  or  $L$  Mbps). For every primary data flow, the primary packet size is fixed to 1500 bytes, the primary message length is fixed to 30 packets and Poisson process message mean arrival rate is set to 1/100 messages/ms. Since the time slot duration is 1 ms, then the mean packet arrival rate per flow is 1/3 packets/ms. Results are averaged over 10 runs, and each run is simulated over a 100 slot (100 ms) time window.

As discussed in Section 3.6.1, Figure 4.1, shows that the higher the delay tolerance of the primary traffic, the more secondary friendly it can be, since it can admit a larger number of secondary calls. Figure 4.1 compares the OFFLINE-VBR-OPT upper bound on the achievable secondary VoIP capacity to the performance of the online VBR schedulers, VBR-VN and VBR-TB. Figure 4.1 also shows the performance of the VBR-TB scheduler.

The figure shows the performance of the VBR-VN scheduler in the case of perfect codec match, i.e., the codec configuration of both virtual and real VoIP calls is G711 with 20 ms packetization, as well as a representative case of the codec mismatch, where the real codec configuration is G711 with 20 ms packetization and the virtual codec configuration is G711 with 30 ms packetization. As discussed in Section 3.6.1, we can see that the performance of the VBR-VN scheduler in the perfect VoIP codec match case is better than that of the VBR-TB scheduler, since the VBR-VN scheduler (perfect codec match case) achieves a secondary VoIP capacity that is equal to that of the upper bound. This is due to the fact that for the short 100 ms simulation time window, the VBR-VN scheduler in the perfect match case reaches the optimum scheduling solution, due to the simplicity of the problem. As shown in Section 3.6.4 and as shown in Figure 4.1, the VBR-VN scheduler in the perfect codec match case performs better than the VBR-VN scheduler in the codec mismatch case. Note that the upper bound and the VBR-VN scheduler (codec perfect match case) saturates at 20 VoIP calls, which is caused by the jitter requirement of the secondary calls. On the other hand, the VBR-VN scheduler (codec mismatch case) saturates at 11 VoIP calls. This is because the VBR-VN scheduler creates friendly spectral hole patterns for a G711 with 30 ms packetization (the virtual codec configuration used for virtual secondary VoIP calls scheduling), while the real secondary VoIP calls that opportunistically share the channel use a G711 with 20 ms packetization (the real codec configuration used for secondary VoIP capacity calculation). Hence at the secondary VoIP capacity saturation point of the VN scheduler (codec mismatch case), the number of spectral holes created over the simulation time is almost half the number of spectral holes actually needed for 20 real secondary G711 VoIP calls with 20 ms packetization, which is the secondary VoIP capacity saturation point of the VN scheduler (codec match case). Therefore in Figure 4.1, the VBR-VN scheduler

(codec mismatch case) saturates at almost half the number of VoIP calls at which the VBR-VN scheduler (perfect match case) saturates. It is clear that under very tight packet delay constraints ( $20 \text{ ms} \leq D_e \leq 40 \text{ ms}$ ), the VBR-TB scheduler and the VBR-VN scheduler (codec mismatch case) are unable to schedule a single secondary VoIP call, while the VBR-VN scheduler (codec perfect match case) and the upper bound are able to admit from 2 to 8 secondary VoIP calls. The VBR-VN scheduler (codec mismatch case) and the VBR-TB scheduler achieve a non-zero secondary VoIP capacity as the primary delay tolerance increases, at  $D_e \geq 50 \text{ ms}$  and  $D_e \geq 200 \text{ ms}$ , respectively. However, the performance of the VBR-TB scheduler improves and surpasses that of the VBR-VN scheduler (codec mismatch case) under relaxed primary delay requirements ( $600 \text{ ms} \leq D_e \leq 1000 \text{ ms}$ ), as shown in Figure 4.1.

#### 4.5.2 Effect of Primary Traffic Load on Secondary VoIP Friendliness

In this section we investigate the effect of changing the primary network traffic load on the secondary VoIP capacity that can be achieved by the online friendly VBR schedulers at the primary basestation. The performance of the VBR-TB and VBR-VN schedulers is studied under varying primary delay tolerance as well. Similar to the simulation model in Section 4.5.1, consider 3 primary Poisson process multi-packet message data arrivals with identical traffic descriptors. Each primary flow is transmitted from the primary basestation to a primary user at a different transmission bit rate ( $H$  or  $M$  or  $L$  Mbps) with a fixed primary packet size of 1500 bytes. However, in this section different levels of the primary network traffic load are studied by varying the mean packet arrival rate per flow. Two simulation parameters control the mean packet arrival rate per flow: the mean message arrival rate per flow, denoted by  $\lambda_p$  and the message length per flow, denoted by  $M_p$ . To

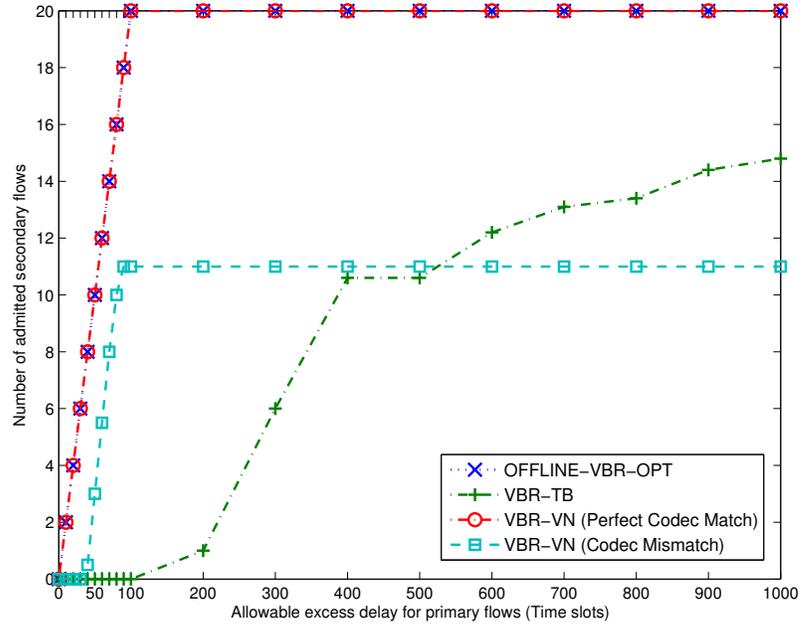


Figure 4.1: Secondary VoIP Capacity vs. Primary Delay Tolerance for OFFLINE-VBR-OPT, VBR-VN and VBR-TB Schedulers. The VBR-VN scheduler is in both perfect codec match and codec mismatch cases. G711-20 ms is used as the real codec configuration. G711-30 ms is used as the virtual codec configuration for the VBR-VN scheduler in the codec mismatch case.

gain better insight into the system performance, we fix one parameter and change the other. All simulations are averaged over 10 runs, and each run is simulated over a 2000 slot (2000 ms) time window.

The performance of the VBR-TB scheduler and the VBR-VN scheduler in both perfect codec match and one representative codec mismatch case, are investigated under varying primary traffic load, i.e, changing  $\lambda_p$  or  $M_p$  and varying primary traffic delay tolerance, i.e., changing  $D_e$ . Note that for a fixed primary packet size, the minimum token rate threshold  $r_{min}$  of the VBR-TB scheduler is set based on the mean primary packet arrival rate averaged over all the primary flows. Hence,  $r_{min}$  is recomputed every experiment based on the  $\lambda_p$  and  $M_p$  values. Similar to the simulation model in Section 4.5.1, the secondary VoIP call

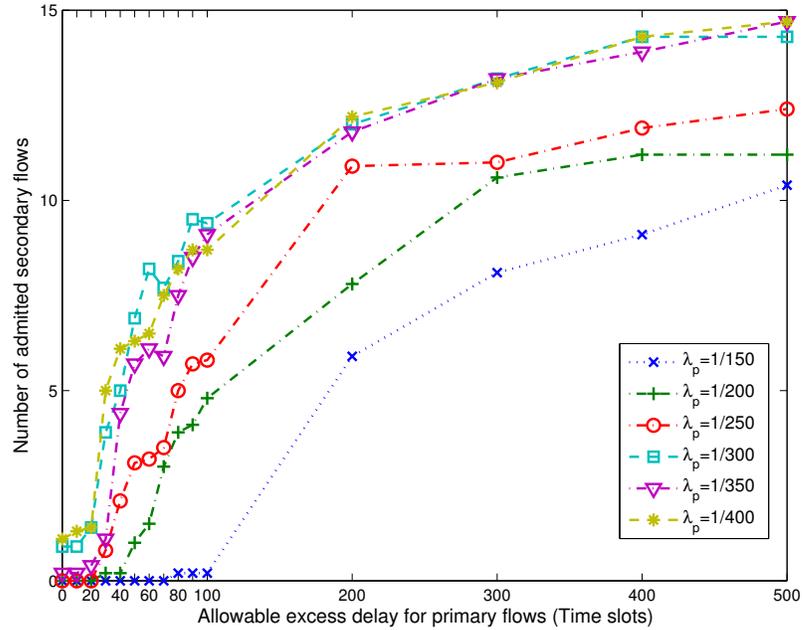


Figure 4.2: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values are used for the mean packet arrival rate ( $\lambda_p$ ) of primary data flows. G711-20 ms is used as the codec configuration.

codec configuration used is G711 with a 20 ms packetization period. G711 with a 30 ms packetization period is used as the virtual VoIP codec configuration for the VBR-VN scheduler in the codec mismatch case.

### Mean Message Arrival Rate

In this section performance results are presented under varying values of mean message arrival rate per primary flow,  $\lambda_p$ . Figures 4.2, 4.3 and 4.4 show the secondary VoIP capacity achieved by the VBR-TB scheduler, the VBR-VN scheduler in the perfect codec match case and the VN scheduler in the codec mismatch case, respectively, for  $M_p = 10$  packets and  $\lambda_p = 1/150, 1/200, 1/250, 1/300, 1/350$  and  $1/400$  messages/ms.

In Figures 4.2, 4.3 and 4.4, it can be seen that  $\lambda_p = 1/150$  messages/ms achieves the

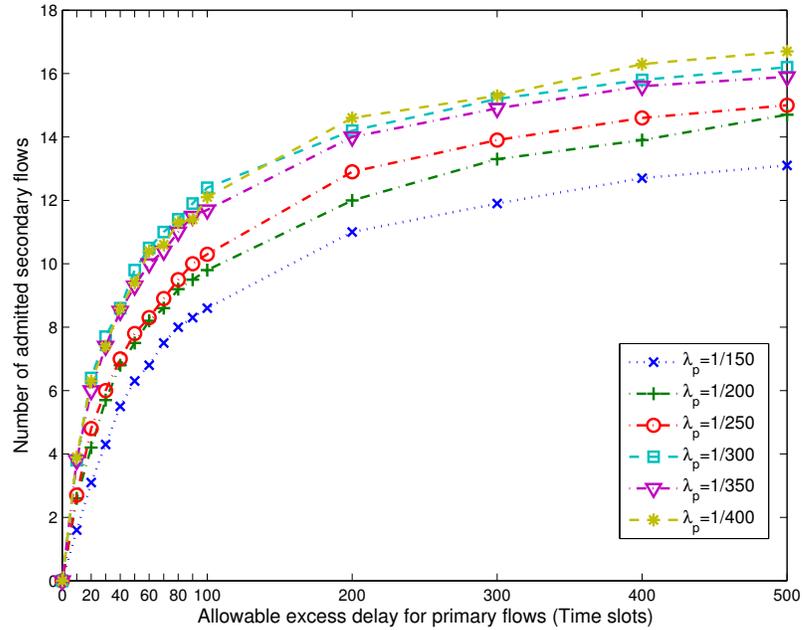


Figure 4.3: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the mean packet arrival rate ( $\lambda_p$ ) of primary data flows. Perfect codec match case (G711-20 ms as the codec configuration)

lowest secondary VoIP capacity, since at this arrival rate per flow, the number of time slots occupied by primary traffic is the largest compared to other tested values of  $\lambda_p$ . This is due to the fact that the higher the mean primary message arrival rate per flow, the lower the secondary VoIP capacity achieved by the proposed schedulers, as shown in Figures 4.2, 4.3 and 4.4. Due to the different transmission bit rates, primary flows transmitted at the lower bit rates have a greater impact on increasing the primary channel utilization.

Comparing Figures 4.2, 4.3 and 4.4 for a given value of  $\lambda_p$ , we can conclude that the performance of the VBR-VN scheduler in the perfect codec match case is generally better than both its performance in the codec mismatch case (with G711-30 ms as the virtual codec used) and the performance of the VBR-TB scheduler, for the same traffic traces. Additionally, for a given value of  $\lambda_p$ , the performance of the VBR-VN scheduler

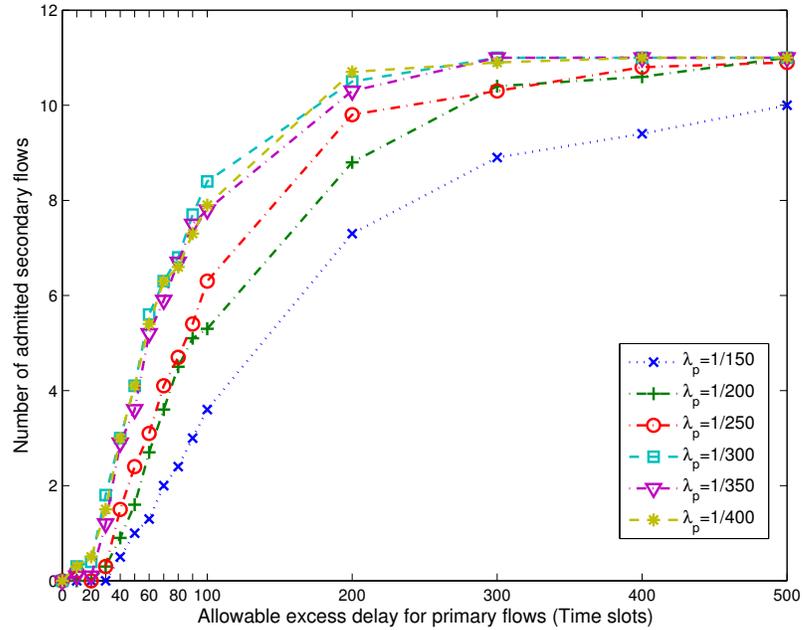


Figure 4.4: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the mean packet arrival rate ( $\lambda_p$ ) of primary data flows. Codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration).

in the codec mismatch case is better than that of the VBR-TB scheduler for  $D_e \leq 100$  ms. However, as the primary delay tolerance increases ( $D_e \geq 200$  ms), the VBR-TB scheduler performs better than the VBR-VN scheduler in the codec mismatch case. Figure 4.2 shows that for tight primary delay tolerance ( $50 \text{ ms} \leq D_e \leq 100 \text{ ms}$ ), the VBR-TB scheduler starts to achieve a non-zero secondary VoIP capacity only when  $\lambda_p \leq 1/200$  messages/ms. On the other hand, Figures 4.3 and 4.4 show that for tight primary delay tolerance ( $50 \text{ ms} \leq D_e \leq 100 \text{ ms}$ ), the VBR-VN scheduler achieves a non-zero secondary VoIP capacity for all values of  $\lambda_p$ , including  $\lambda_p = 1/150$  messages/ms. We can also conclude from the three figures that, the impact of changing the mean primary message arrival rate on the achieved secondary VoIP capacity decreases as the value of the mean message arrival rate decreases. Therefore, the secondary VoIP capacity achieved by a given proposed VBR scheduler is

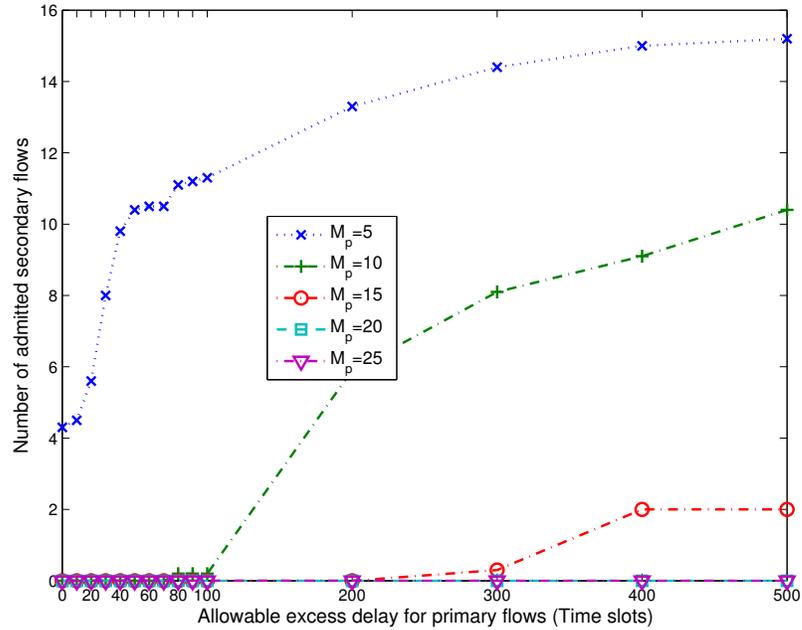


Figure 4.5: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values are used for the message length ( $M_p$ ) of primary data flows. G711-20 ms is used as the codec configuration.

almost the same for  $\lambda_p = 1/300, 1/350$  and  $1/400$  messages/ms. This can be considered a saturation point after which any further decrease in the mean message arrival rate does not further increase the secondary VoIP capacity achieved by the proposed schedulers.

### Primary Message Length

In this section performance results are presented under varying values of primary message length per primary flow,  $M_p$ . Figures 4.5, 4.6 and 4.7 show the secondary VoIP capacity achieved by the VBR-TB scheduler, the VBR-VN scheduler in the perfect codec match case and the VN scheduler in the codec mismatch case, respectively, for  $\lambda_p = 1/150$  messages/ms and  $M_p = 5, 10, 15, 20$  and 25 packets. Figures 4.5, 4.6 and 4.7 can be analyzed in the same fashion as Figures 4.2, 4.3 and 4.4.

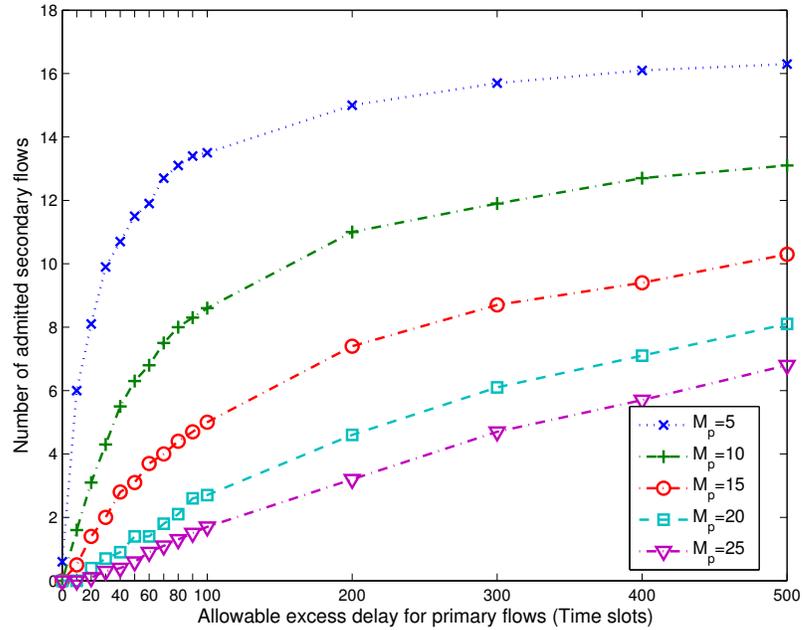


Figure 4.6: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the message length ( $M_p$ ) of primary data flows. Perfect codec match case (G711-20 ms as the codec configuration).

In Figures 4.5, 4.6 and 4.7, it can be seen that  $M_p = 25$  packets achieves the lowest secondary VoIP capacity, since at this primary message length per flow, the number of time slots occupied by primary traffic is the largest compared to other tested values of  $M_p$ . This is due to the fact that the higher the primary message length per flow, the lower the secondary VoIP capacity achieved by the proposed schedulers, as shown in Figures 4.5, 4.6 and 4.7. Due to the different transmission bit rates, primary flows transmitted at the lower bit rates have a greater impact on increasing the primary channel utilization.

Comparing Figures 4.5, 4.6 and 4.7 for a given value of  $M_p$ , we can conclude that the performance of the VBR-VN scheduler in the perfect codec match case is generally better than both its performance in the codec mismatch case (with G711-30 ms as the virtual codec used) and the performance of the VBR-TB scheduler, for the same traffic traces.

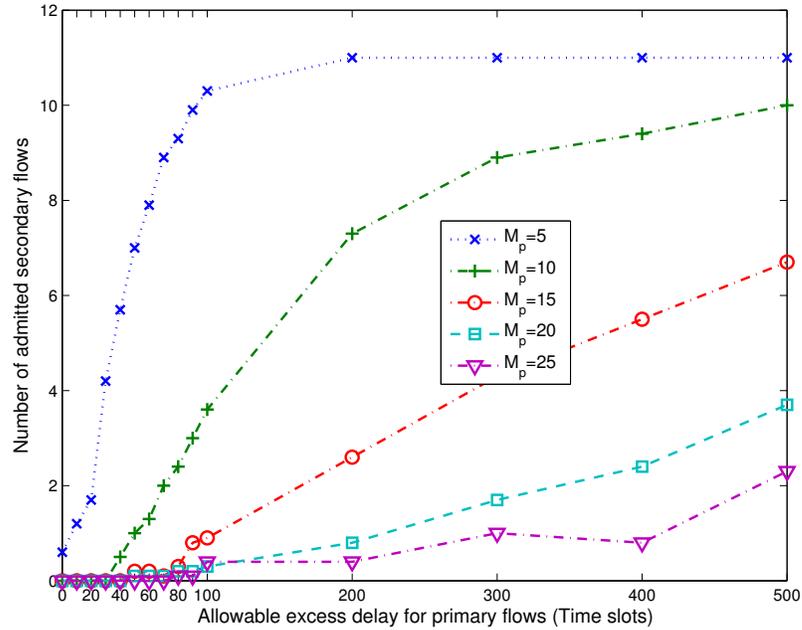


Figure 4.7: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values are used for the message length ( $M_p$ ) of primary data flows. Codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration).

Additionally, for  $M_p \geq 10$  packets, the performance of the VBR-VN scheduler in the codec mismatch case is better than that of the VBR-TB scheduler, under varying values of primary delay tolerance. However, for  $M_p = 5$  packets, the VBR-TB scheduler performs better than the VBR-VN scheduler in the codec mismatch case. The performance of the VBR-TB scheduler is critically affected as the primary message length increases. This is shown in Figure 4.5, where the VBR-TB scheduler achieves a zero secondary VoIP capacity under different levels of primary delay tolerance, for  $M_p \geq 20$  packets. This is due to the fact that primary message lengths that are larger than the secondary VoIP call jitter requirement tend to block the channel for consecutive time slots, making it more difficult for VoIP calls to coexist. However, the VN scheduler (perfect codec match), avoids this problem using the virtual node scheduling technique and achieves a non zero secondary VoIP capacity for

$M_p \geq 20$  packets under different levels of primary delay tolerance, as shown in Figure 4.6. As shown in Figure 4.7, the VN scheduler (codec mismatch case) achieves a non zero secondary VoIP capacity for  $M_p \geq 20$  packets under non-tight primary delay tolerance ( $100 \text{ ms} \leq D_e \leq 1000 \text{ ms}$ ). Figure 4.5 also shows that for tight primary delay tolerance ( $0 \text{ ms} \leq D_e \leq 100 \text{ ms}$ ), the VBR-TB scheduler achieves a non-zero secondary VoIP capacity only at  $M_p = 5$  packets. On the other hand, Figures 4.6 and 4.7 show that for tight primary delay tolerance ( $0 \text{ ms} \leq D_e \leq 100 \text{ ms}$ ), the VBR-VN scheduler (perfect codec match case) achieves a non-zero secondary VoIP capacity for all values of  $M_p$ , while the VBR-VN scheduler (codec mismatch case) starts to achieve a non-zero secondary VoIP capacity when  $M_p \leq 10$  packets. We can also conclude from the three figures that the impact of changing the primary message length on the achieved secondary VoIP capacity decreases as the value of the message length increases. Therefore, the secondary VoIP capacity achieved by a given proposed VBR scheduler is relatively close for  $M_p = 20$  and 25 packets.

## 4.6 Conclusions

In this chapter we have proposed primary basestation scheduling mechanisms that are designed to be friendly from a secondary network user perspective. Multiple primary users were considered, each receiving data packets subject to delay constraints from the primary basestation at variable bit rates (VBR). As in Chapter 3, the performance of the proposed friendly scheduling at the primary basestation is measured using the secondary VoIP capacity metric, defined in Chapter 3. We first developed an offline VBR scheduler, OFFLINE-VBR-OPT, which maximizes friendliness over finite time intervals. Then we presented two online friendly VBR schedulers, VBR-TB and VBR-VN, based on spectral hole shaping and virtual node scheduling, respectively. The online friendly VBR schedulers are

modified versions of the previously proposed schedulers, TB and VN, to support multiple primary data flows with variable transmission bit rates. Simulation results are presented that show the effect of the primary traffic load and primary network delay tolerance on the level of the primary network friendliness to potential secondary VoIP traffic.

The VBR-TB scheduler uses an adaptive token rate setting scheme to adjust the spectral hole shaping based on the current state of the primary traffic backlog, aggregated from multiple primary traffic flows. This is done in a way that creates spectrum holes as friendly as possible while considering different transmission rates of different primary flows. A minimum token rate threshold avoids future primary QoS violation based on past scheduling decisions, which guarantees feasible primary traffic scheduling as the primary traffic burstiness increases. Experimenting with different values, we concluded that the mean primary packet arrival rate averaged over multiple primary flows, is a good candidate value for the minimum token rate threshold. Considering different transmission rates of different primary users, the VBR-VN scheduler schedules the aggregated real primary traffic along with virtual secondary VoIP calls. As concluded from the performance results in Chapter 3, the performance of the VN scheduler largely depends on the choice of the codec configuration of the virtual and real VoIP traffic, which applies to the VBR-VN scheduler.

As in Chapter 3, the results showed that for both the proposed schedulers and the offline scheduling bound, the higher the delay tolerance of the primary traffic, the more secondary friendly it can be. Also as expected, both of the proposed schedulers achieved lower performance than the non-causal friendliness upper bound. Considering primary data flows with identical traffic descriptors, we experimented varying the mean primary message arrival rate and the primary message length. It was shown that for a given value of message

length or mean message arrival rate, the performance of the VBR-VN scheduler in the perfect codec match case was generally better than both its performance in the codec mismatch case and the performance of the VBR-TB scheduler. In the codec mismatch case, the VBR-VN scheduler achieved performance that is better than that of the VBR-TB scheduler, under relatively tight primary delay requirements. However, under relaxed primary delay requirements, the VBR-TB scheduler performed better than the VBR-VN scheduler in the codec mismatch case with some restrictions on the message length and mean message arrival rate values. Hence, we concluded that under tight primary delay requirements, the VBR-TB online friendly scheduler is not the scheduler of choice. Results also showed that the impact of changing the mean primary message rate on the achieved secondary VoIP capacity decreases as the value of the mean message arrival rate decreases. Similarly, the impact of changing the primary message length on the achieved secondary VoIP capacity decreases as the value of the message length increases. Primary message lengths that are larger than the secondary VoIP call jitter requirement tended to block the channel for consecutive time slots, making it more difficult for VoIP calls to coexist, which resulted in low secondary VoIP capacity. Our results showed that the changing the primary network load has a larger impact on the VBR-TB scheduler, to the extent that it achieved a zero VoIP capacity for certain values of primary message length and mean message arrival rate, especially under tight primary delay requirements.

# Chapter 5

## Relay Node Placement for Secondary VoIP Friendliness

### 5.1 Introduction

In this chapter we propose a framework to reduce the primary channel utilization and increase the capacity available for opportunistic spectrum sharing by secondary users. A set of fixed relay nodes is placed within full coverage of the primary basestation and primary users. Using the decode-and-forward relaying protocol, the relay nodes relay the primary data flows from the primary basestation to the primary users. This enhances the primary basestation transmission rates. From the secondary network point of view, the cooperation between the primary basestation and relay nodes is translated into more spectrum holes and higher secondary traffic capacity. Our previously proposed online friendly VBR schedulers are then used to shape the enhanced spectrum hole pattern to be friendly to potential secondary VoIP calls.

We first present a formulation of the offline relay selection problem, whose solution is

a lower bound on the achievable primary channel utilization. An integer linear program is introduced that can be solved to find the set of optimal relay locations that minimize the number of time slots occupied by primary traffic. A greedy offline relay placement (GORP) algorithm is then introduced that provides a relay selection decision by making greedy mappings of primary data flows to relay nodes. The proposed GORP algorithm is compared against the lower bound for different numbers of available relay nodes. Three representative primary user layouts are considered that show different levels of achievable primary channel utilization under the proposed cooperative framework. For each layout, the secondary VoIP capacity achieved by our online friendly schedulers, VBR-TB and VBR-VN, is investigated. Our results show that the number of open cooperative relay nodes directly affects the secondary friendliness. The results also show that the effect of the proposed cooperative framework on the secondary VoIP capacity depends on the primary user layout and the number of open relays.

The remainder of the chapter is organized as follows. Section 5.2 summarizes some of the related work that studies the reduction of the primary channel utilization under the cooperative spectrum sharing model. Section 5.3 describes the assumptions used in our model. Section 5.4 presents the proposed offline relay placement solutions and the (integer linear program) ILP formulation of the problem is presented in Section 5.4.1. Section 5.4.2 introduces the GORP algorithm. Following this, in Section 5.5 the simulation model and results are presented. In Section 5.5.1 the offline relay placement solutions of the bound and the GORP algorithm are studied. Section 5.5.2 then shows the secondary VoIP capacity achieved by the online friendly schedulers for different levels of primary channel utilization. Finally, Section 5.6 presents the conclusions of this chapter.

## 5.2 Related Work

In addition to the conventional *interweave* spectrum sharing model, many recent research efforts have studied *overlay* spectrum sharing. In *overlay* spectrum sharing, the secondary users use signal processing and coding to maintain or improve the communication of primary users while also obtaining some additional bandwidth for their own communication [10]. This cooperation framework allows higher transmission rates for primary users and higher capacity for secondary users, which result in improved performance with incentives for both networks to cooperate. Applying cooperative communication within a cognitive radio network results in a *cooperative cognitive radio network (CCRN)*, where secondary users or access points are leveraged to relay primary traffic. The CCRN model relaxes the assumption of oblivious primary users in the original cognitive radio network definition [11], which, in turn, allows various forms of cooperation. In this section we focus on some research efforts that study the CCRN spectrum leasing model of cooperative primary and secondary users, utilizing the time domain decode-and-forward (DF) relaying technique.

In [5], a DF cooperative spectrum leasing solution was proposed based on the idea that secondary nodes can earn spectrum access in exchange for cooperation with the primary link. In [90], a cooperative communication-aware spectrum leasing framework is proposed for collocated infrastructure-based primary and multiple secondary networks. In the proposed spectrum leasing model, the primary network uses the secondary APs as cooperative relays. The primary network decides the optimal strategy for the relay selection and the price for spectrum leasing, and the secondary networks determine the length of spectrum access time to purchase from the primary network. Simulation results show that the primary network and secondary networks achieve higher utilities by exploiting cooperative

transmission.

The purpose of this chapter is to optimally place a set of fixed relay nodes within the primary network coverage area. The relays cooperate with the primary basestation to reduce the primary channel utilization. The concept of cooperation between the primary basestation and the relay nodes resembles that of the CCRN spectrum leasing models. This chapter studies the performance from the secondary VoIP friendliness point of view.

### 5.3 System Model

Two collocated infrastructure based primary and secondary networks are considered that share a single licensed channel owned by the primary network, based on an *interweave* spectrum sharing model [10]. A time-invariant channel model is assumed, where variable bit rate transmission is used at the primary stations. Multiple primary users receive multi-packet message data from the primary basestation on the channel in question. A set of fixed relay nodes cooperatively relay the primary traffic from the primary basestation to the primary users.

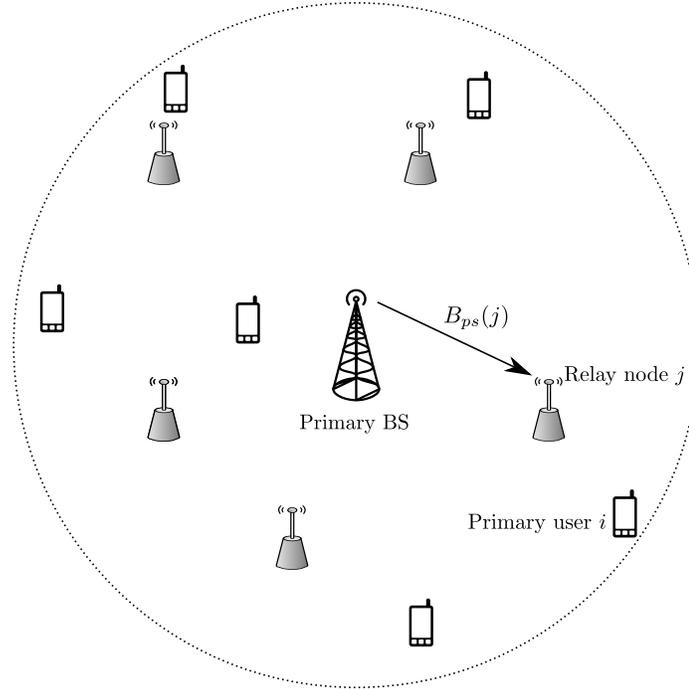
A primary transmission can occur in either direct transmission mode or cooperate-to-join mode. In direct transmission mode, the primary user directly receives its data from the primary basestation. In the cooperate-to-join mode, fixed relay nodes will cooperate with the primary basestation by relaying the primary downlink traffic to the primary users. It is assumed that decode-and-forward (DF) is adopted as the cooperation strategy [5, 89, 90], where maximum ratio combining is applied at the primary user [90]. A given primary user can be associated with one cooperative relay node.

The primary users communicate with their basestation using a demand TDMA protocol. Since the channel is accessed in TDMA mode, the same cooperative relay node can

be associated with different primary users. The time axis is divided into periodic super-frame intervals, each of which consists of a fixed number of time slots used for packet data. Each primary user's time slot is divided into two subslots that represent two phases of the cooperation process. As an example, consider the two-phase cooperation scenario shown in Figures 5.1a and 5.2a. In the first cooperation phase, the primary basestation transmits the primary data flow to its assigned relay, as shown in Figure 5.1a. In the second cooperation phase, the primary basestation and the relay cooperatively transmit to the primary user, which performs maximum ratio combining at its end, as shown in Figure 5.2a. This cooperation process increases the effective downlink bit rate, leading to reduced channel utilization. For simplicity, the ACK transmission is not shown in Figures 5.1a and 5.2a. However, it is assumed that at the end of every primary transmission, the primary basestation receives an ACK from the primary user, which could be received on a separate uplink channel using either cooperative or direct transmission [90]. We assume the same secondary network model and packet deadline assignment as in Section 3.3. Our objective is to optimally place a set of fixed relay nodes that will maximize the overall primary data rate enhancement. The inputs and outputs of our problem are more formally defined as follows.

**INPUT:** A set of candidate locations is defined for the relay nodes as  $\mathcal{R}_s$ . We define  $\mathcal{F}_p$  as the input set of primary data flows. The maximum number of relay nodes that can be deployed is denoted as  $R_{max}$ .  $\mathcal{T}$  represents the set of time slots over which the primary packet scheduling and relay placement occur, while  $\sigma$  represents the duration of a single time slot.

Define  $B(i, j)$  as the overall achievable data rate when a relay node at location



(a) Primary basestation transmit data of primary user  $i$  to relay node  $j$ , at bit rate  $B_{ps}(j)$ .

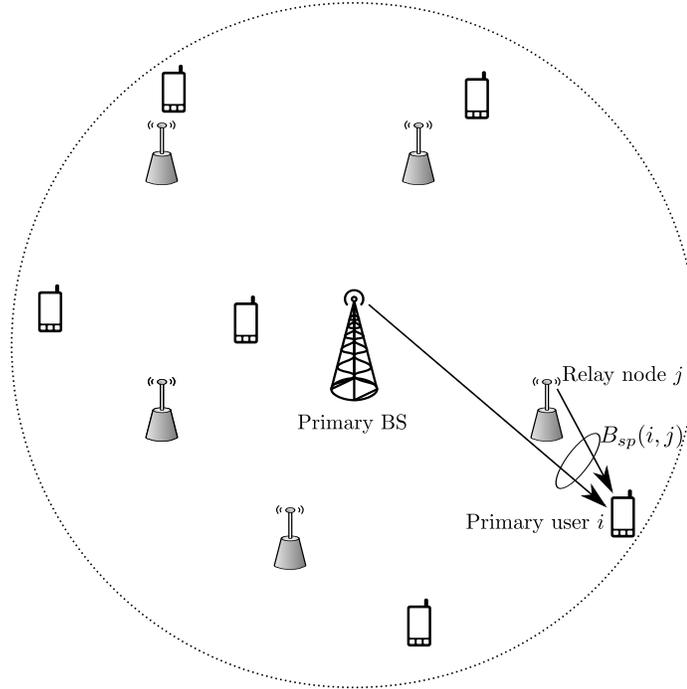


(b) The first phase occupies the first fraction  $(1 - \alpha(i, j))$  of the time slot.

Figure 5.1: First Cooperation Phase Scenario

$j \in \mathcal{R}_s$  relays the data of primary flow  $i \in \mathcal{F}_p$ . This is calculated as the minimum achieved bit rate in each cooperation phase, taking into consideration the time division ratio between the two phases [90]. Define  $\alpha(i, j)$  as the parameter dividing the time slot between the two cooperation phases when relay node location  $j \in \mathcal{R}_s$  relays data of primary flow  $i \in \mathcal{F}_p$ , as shown in Figures 5.1b and 5.2b.

Assume  $j^* \in \mathcal{R}_s$  represents a fictitious relay node location that does not provide any primary traffic relaying. Hence  $B(i, j^*)$  would be equal to  $B_p(i)$ , which is the



(a) Primary basestation and relay node  $j$  cooperatively transmit data to primary user  $i$ , at bit rate  $B_{sp}(i, j)$  after maximum ratio combining.



(b) The second phase occupies the remaining fraction  $\alpha(i, j)$  of the time slot.

Figure 5.2: Second Cooperation Phase Scenario

data rate achieved when primary user  $i$  directly receives its data from the primary basestation with no relaying.  $B_p(i)$  is calculated as follows:  $B_p(i) = \log_2(1 + \eta_p(i))$ , where  $\eta_p(i)$  is the SNR of the received signal at primary user  $i$  from the primary basestation. Then  $B(i, j)$  is computed as follows:

$$B(i, j) = \begin{cases} \min((1 - \alpha^*(i, j))B_{ps}(j), \alpha^*(i, j)B_{sp}(i, j)) & j \neq j^* \\ B_p(i) & j = j^* \end{cases} \quad (5.1)$$

where  $\alpha^*(i, j)$  represents the optimal value for  $\alpha(i, j)$  based on equation (12) in [90], calculated as follows:

$$\alpha^*(i, j) = \frac{B_{ps}(j)}{B_{ps}(j) + B_{sp}(i, j)} \quad (5.2)$$

where  $B_{ps}(j)$  is the bit rate achieved in the first cooperation phase, as shown in Figure 5.1a. It is calculated as follows:  $B_{ps}(j) = \log_2(1 + \eta_{ps}(j))$ , where  $\eta_{ps}(j)$  is the SNR of the received signal at relay  $j$  from the primary basestation.  $B_{sp}(i, j)$  is the bit rate achieved using maximum ratio combining in the second cooperation phase, as shown in Figure 5.2a. It is calculated as follows:  $B_{sp}(i, j) = \log_2(1 + \eta_{sp}(i, j) + \eta_p(i))$  where  $\eta_s(i, j)$  is the SNR of the received signal at primary user  $i$  from relay  $j$  and  $\eta_p(i)$  is the SNR of the received signal at primary user  $i$  from the primary basestation. Hence, Equation 5.1 ensures that the achievable data rate when relay  $j$  relays data of primary flow  $i$ , does not exceed the minimum achieved bit rate in each phase of the cooperation. Moreover, it implies that the achievable cooperative bit rate obeys the channel capacity estimation rules used in calculating  $B_{ps}(j)$  and  $B_{sp}(i, j)$ .

Define  $\mathcal{U}_p(i)$  to be the set of packets of primary flow  $i$ . Every primary packet,  $p \in \mathcal{U}_p(i)$ , has a packet length in bits  $L_p(p, i)$ , an arrival time and a delay constraint, and we define

$$\mathcal{T}_p(p, i) = \left\{ \begin{array}{l} t \in \mathcal{T} \mid \text{time slot } t \text{ satisfies the QoS constraints for primary} \\ \text{packet } p \text{ that belongs to the primary data flow } i. \end{array} \right\} \quad (5.3)$$

i.e.,  $\mathcal{T}_p(p, i) \subset \mathcal{T}$  is the set of time slots, any of which would satisfy packet  $p$ 's

QoS constraints if assigned to that packet. Since the primary traffic may have delay tolerance, it is allowed that a fraction of packets,  $\epsilon$ , can be dropped with no scheduling when their delay constraints are violated.

**OUTPUT:** The objective is to place a set of relay nodes in a way that maximizes the overall achieved primary data rate. More formally, we define

$$x(i, j) = \begin{cases} 1 & \text{if primary data flow } i \text{ is being relayed by a relay node at location} \\ & j, \\ 0 & \text{otherwise.} \end{cases} \quad (5.4)$$

for all  $i \in \mathcal{F}_p$  and  $j \in \mathcal{R}_s$ .

$$y(j) = \begin{cases} 1 & \text{if a relay node is deployed at location } j, \\ 0 & \text{otherwise.} \end{cases} \quad (5.5)$$

for all  $j \in \mathcal{R}_s, j \neq j^*$

$$A(p, i, j, t) = \begin{cases} 1 & \text{if primary packet } p \text{ of primary data flow } i \text{ is relayed by relay } j \\ & \text{and transmitted at time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (5.6)$$

for all  $p \in \mathcal{U}_p(i), t \in \mathcal{T}_p(p, i), i \in \mathcal{F}_p$  and  $j \in \mathcal{R}_s$

$$d_p(p, i) = \begin{cases} 1 & \text{if primary packet } p \text{ belonging to primary data flow } i \text{ is dropped,} \\ 0 & \text{otherwise.} \end{cases} \quad (5.7)$$

for all  $p \in \mathcal{U}_p(i)$  and  $i \in \mathcal{F}_p$ . Given these definitions and the above inputs, the objective is to assign values for the decision variables defined above that maximize the overall achieved primary data rate.

## 5.4 Offline Relay Node Placement

In this section we first formulate an optimization that computes a lower bound on the primary channel utilization achieved by the proposed cooperative communication. The optimal offline schedule is derived using an integer linear program (ILP) whose solution has worse-case exponential time complexity. To be able to handle practical size problems, we then propose a greedy offline relay placement algorithm (GORP algorithm). The GORP algorithm is later compared to the bound performance from the primary channel utilization point of view, in Section 5.5.1.

### 5.4.1 Optimal Offline Relay Placement

The problem objective is to optimally open the best candidate relay nodes that will minimize the number of time slots occupied by the primary traffic, constrained by the maximum number of open relay nodes and the QoS requirements of the primary network. The bound is obtained by taking a complete time window of primary inputs. As an input, the formulated ILP takes a given set of primary traffic traces, primary flow direct transmission

rates, cooperative transmission rates and the maximum number of open relay nodes,  $R_{max}$ . As an output, assignments are made for  $x(i, j)$ ,  $y(j)$ ,  $A(p, i, j, t)$  and  $d_p(p, i)$  defined in Section 5.3. Using the definitions from Section 5.3, the optimization problem, denoted by OFFLINE-RP-OPT, is as follows.

$$\text{minimize} \quad \sum_{p \in \mathcal{U}_p(i)} \sum_{i \in \mathcal{F}_p} \sum_{j \in \mathcal{R}_s} \sum_{t \in \mathcal{T}_p(p, i)} A(p, i, j, t) \left[ \frac{L_p(p, i)}{\sigma B(i, j)} \right] \quad (\text{OFFLINE-RP-OPT})$$

subject to

$$\sum_{i \in \mathcal{F}_p} \sum_{p \in \mathcal{U}_p(i)} \sum_{j \in \mathcal{R}_s} A(p, i, j, t) \leq 1, \quad \forall t \in \mathcal{T} \quad (5.8)$$

$$x(i, j) \leq \sum_{p \in \mathcal{U}_p(i)} \sum_{t \in \mathcal{T}_p(p, i)} A(p, i, j, t), \quad \forall i \in \mathcal{F}_p, j \in \mathcal{R}_s \quad (5.9)$$

$$x(i, j) \geq A(p, i, j, t), \quad \forall i \in \mathcal{F}_p, p \in \mathcal{U}_p(i), j \in \mathcal{R}_s, t \in \mathcal{T} \quad (5.10)$$

$$\sum_{t \in \mathcal{T}_p(p, i)} \sum_{j \in \mathcal{R}_s} A(p, i, j, t) B(i, j) \geq (1 - d_p(p, i)) L_p(p, i), \quad \forall p \in \mathcal{U}_p(i), i \in \mathcal{F}_p \quad (5.11)$$

$$\sum_{p \in \mathcal{U}_p(i)} d_p(p, i) \leq \epsilon |\mathcal{U}_p(i)|, \quad \forall i \in \mathcal{F}_p \quad (5.12)$$

$$\sum_{j \in \mathcal{R}_s} x(i, j) \leq 1, \quad \forall i \in \mathcal{F}_p \quad (5.13)$$

$$\sum_{j \in \mathcal{R}_s, j \neq j^*} y(j) \leq R_{max}, \quad (5.14)$$

$$y(j) \geq x(i, j), \quad \forall i \in \mathcal{F}_p, \quad \forall j \in \mathcal{R}_s, j \neq j^* \quad (5.15)$$

$$y(j) \leq \sum_{i \in \mathcal{F}_p} x(i, j), \quad \forall j \in \mathcal{R}_s, j \neq j^* \quad (5.16)$$

$$A(p, i, j, t) \in \{0, 1\}, \quad \forall p \in \mathcal{U}_p(i), i \in \mathcal{F}_p, j \in \mathcal{R}_s, t \in \mathcal{T} \quad (5.17)$$

$$d_p(p, i) \in \{0, 1\}, \quad \forall p \in \mathcal{U}_p(i), i \in \mathcal{F}_p \quad (5.18)$$

$$x(i, j) \in \{0, 1\}, \quad \forall i \in \mathcal{F}_p, j \in \mathcal{R}_s \quad (5.19)$$

$$y(j) \in \{0,1\}, \quad \forall j \in \mathcal{R}_s, j \neq j^* \quad (5.20)$$

The objective of OFFLINE-RP-OPT is to minimize the total number of conforming time slots needed to transmit all primary packets of all primary flows, using either direct transmission or cooperative relaying. Constraint (5.8) forces the single channel assumption, i.e., any given time slot is assigned to only one primary packet. Constraints (5.9) and (5.10) set the proper relationship between the two decision variables  $x(i, j)$  and  $A(p, i, j, t)$  to ensure that when relay  $j$  helps relaying primary packet  $p \in \mathcal{U}_p(i)$  belonging to primary user  $i$ , then relay  $j$  must be assigned to primary user  $i$ . Constraint (5.11) ensures that scheduled primary packets meet their deadline constraints, while constraint (5.12) ensures that the number of primary packets dropped per primary transmission does not exceed the maximum allowable dropping threshold. Constraint (5.13) ensures that every primary transmission is associated with only one deployed relay node. Constraint (5.14) guarantees that the total number of deployed relay nodes does not exceed the maximum allowed. Constraints (5.15) and (5.16) set the relationship between the decision variables,  $x(i, j)$  and  $y(j)$ , to ensure that a relay node will be deployed at a given candidate location  $j$  (i.e.,  $y(j) = 1$ ), if this location is used to relay at least one primary flow data. Constraints (5.17), (5.18), (5.19) and (5.20) define the binary decision variables.

The ILP formulated above can be relaxed to a less complex conventional linear program by removing the binary constraints on the decision variables. However, the formulated ILP is solvable for short simulation time intervals as shown later in Section 5.5.1. Although the ILP formulated above provides both a primary traffic schedule and an offline relay placement decision, in this chapter we are interested in the latter part of the solution.

### 5.4.2 Greedy Offline Relay Placement (GORP) Algorithm

In this section we present a greedy offline relay placement algorithm. To maximize the primary achievable data rates and minimize the primary channel utilization, the GORP algorithm sorts the primary flows based on their priority for helping by a relay node, which is determined according to the traffic intensity of the primary flows. The traffic intensity,  $\rho_p(i)$ , of a given primary flow  $i \in \mathcal{F}_p$  is defined as a function of the primary flow direct transmission rate and the primary flow traffic load as follows:

$$\rho_p(i) = \frac{\overline{L}_p(i)\lambda_p(i)}{B_p(i)} \quad (5.21)$$

where  $\overline{L}_p(i)$  is the mean packet length in bits of primary flow  $i$ ,  $\lambda_p(i)$  its mean packet arrival rate in packets/second.  $B_p(i)$  is the direct transmission rate in bits/second from the primary basestation to primary user  $i$ .

Equation (5.21) computes the exact traffic intensity of a given primary flow  $i$ , based on the direct transmission rate  $B_p(i)$  in bits/second, however the GORP algorithm is more interested in the number of time slots a given primary flow  $i$  needs to transmit a single packet, using the direct transmission rate  $B_p(i)$ , which would be  $\left\lceil \frac{\overline{L}_p(i)}{\sigma B_p(i)} \right\rceil$ , where  $\sigma$  is the duration of a single time slot in seconds. Based on this, we redefine the traffic intensity of primary flow  $i$  as follows:

$$\rho_p(i) = \left\lceil \frac{\overline{L}_p(i)}{\sigma B_p(i)} \right\rceil \sigma \lambda_p(i) \quad (5.22)$$

A formal description of the algorithm is shown in Algorithm 1. Steps 1 and 2 initialize the solution variables, defined in Section 5.3,  $x(i, j)$  for all  $i \in \mathcal{F}_p$  and  $y(j)$  for all  $j \in \mathcal{R}_s, j \neq j^*$ , respectively. Step 3 initializes the set of open relay nodes,  $\mathcal{R}_s^{open}$ , to an

**Algorithm 1** Greedy Offline Relay Placement (GORP) Algorithm

$\mathcal{F}_p$  : set of primary flows  
 $\mathcal{R}_s$  : set of all candidate relay nodes  
 $y(j)$  : set of binary variables, 1 if relay node  $j$  is opened, 0 otherwise,  $\forall j \in \mathcal{R}_s, j \neq j^*$   
 $x(i, j)$  : set of binary variables, 1 if primary flow  $i$  is relayed by relay  $j$ , 0 otherwise,  $\forall i \in \mathcal{F}_p, \forall j \in \mathcal{R}_s$   
 $B(i, j)$  : achievable cooperative transmission rate in bits/second of primary flow  $i$ , if relayed by relay  $j$ ,  $\forall i \in \mathcal{F}_p, \forall j \in \mathcal{R}_s$   
 $B_p(i)$  : direct transmission rate of primary flow  $i$ ,  $B_p(i) = B(i, j^*), \forall i \in \mathcal{F}_p$   
 $\lambda_p(i)$  : mean packet arrival rate in packets/second of primary flow  $i$ ,  $\forall i \in \mathcal{F}_p$   
 $\bar{L}_p(i)$  : mean packet length in bits of primary flow  $i$ ,  $\forall i \in \mathcal{F}_p$   
 $\sigma$  : duration of a single time slot in seconds  
 $R_{max}$  : maximum allowable number of open relays  
 $\mathcal{R}_s^{open}$  : set of open relay nodes  
 $B_p^{greedy}(i)$  : achievable transmission rate of primary flow  $i$ , direct or cooperative,  $\forall i \in \mathcal{F}_p$

- 1: Initialize  $x(i, j) = 0, \forall i \in \mathcal{F}_p, \forall j \in \mathcal{R}_s$ .
- 2: Initialize  $y(j) = 0, \forall j \in \mathcal{R}_s, j \neq j^*$ .
- 3: Initialize  $\mathcal{R}_s^{open} = \{\}$ .
- 4: Compute  $\rho_p(i) = \left\lceil \frac{\bar{L}_p(i)}{\sigma B_p(i)} \right\rceil \sigma \lambda_p(i), \forall i \in \mathcal{F}_p$
- 5: Let  $\mathcal{F}_p^{sorted}$  be the set of primary flows,  $\mathcal{F}_p$ , sorted in a descending order acc. to  $\rho_p$ .
- 6: **for all**  $i \in \mathcal{F}_p^{sorted}$  **do**
- 7:   **if**  $\sum_{j \in \mathcal{R}_s, j \neq j^*} y(j) < R_{max}$  **then**
- 8:     ASSIGN-BEST-RELAY( $\mathcal{R}_s, B, i, x, y$ ).
- 9:   **else**
- 10:     **if**  $\mathcal{R}_s^{open} = \{\}$  **then**  $\mathcal{R}_s^{open} = \{j \in \mathcal{R}_s, j \neq j^* | y(j) = 1\}$  **end if**
- 11:     ASSIGN-BEST-RELAY( $\mathcal{R}_s^{open}, B, i, x, y$ ).
- 12:   **end if**
- 13:   Compute  $B_p^{greedy}(i) = \sum_{i \in \mathcal{F}_p} \sum_{j \in \mathcal{R}_s} x(i, j) B(i, j)$ .
- 14: **end for**
- 15: **function** ASSIGN-BEST-RELAY( $\mathcal{R}'_s, B, i, x, y$ )
- 16:   Define  $j' = \underset{j \in \mathcal{R}'_s, j \neq j^*}{\operatorname{argmin}} \left\lceil \frac{\bar{L}_p(i)}{\sigma B(i, j)} \right\rceil$ .
- 17:   **if**  $\left\lceil \frac{\bar{L}_p(i)}{\sigma B(i, j')} \right\rceil < \left\lceil \frac{\bar{L}_p(i)}{\sigma B(i, j^*)} \right\rceil$  **then**
- 18:     Assign primary flow  $i$  to relay node  $j'$ , i.e.,  $x(i, j') = 1, y(j') = 1$ .
- 19:   **else**
- 20:     Assign primary flow  $i$  to direct transmission, i.e.,  $x(i, j^*) = 1$ .
- 21:   **end if**
- 22:   **return**  $x, y$
- 23: **end function**

empty set. Step 4 computes the traffic intensity values of all the primary flows,  $\rho_p(i)$  for all  $i \in \mathcal{F}_p$ , based on equation (5.22). Step 5 sorts the primary flows in descending order based on their traffic intensity values. The algorithm gives a higher priority to a primary flow with high traffic intensity. Similar to OFFLINE-RP-OPT, the GORP algorithm aims to minimize the total number of time slots occupied by the primary traffic. In Steps 6 to 14, the algorithm iterates over the primary flows sorted in a descending order according to their traffic intensity. For every primary flow  $i \in \mathcal{F}_p^{sorted}$ , the algorithm calls the ASSIGN-BEST-RELAY function. This function is declared in Steps 15 to 23 and takes as an input the set of available relay nodes  $\mathcal{R}'_s$ , the direct and cooperative transmission rates  $B$ , the index of the current primary flow being served  $i$  and the current values of the solution variables  $x$  and  $y$ . ASSIGN-BEST-RELAY assigns the primary flow  $i$  to its best available cooperative transmission rate and the values of  $x$  and  $y$  are updated accordingly. In Steps 16 and 17, ASSIGN-BEST-RELAY compares the direct transmission rate of primary flow  $i$  and the available cooperative transmission rates. Then primary flow  $i$  is assigned to a candidate relay node,  $j \in \mathcal{R}'_s$  and  $j \neq j^*$ , that provides the best achievable cooperative transmission rate among all options,  $\mathcal{R}'_s$ . Step 18 ensures that a primary flow is transmitted in cooperate-to-join mode only if the best available cooperative rate is better than the primary flow's direct transmission rate. Otherwise, in Steps 19 and 20 the primary flow is set to be transmitted in the direct transmission mode. Step 7 checks if the maximum number of relay nodes,  $R_{max}$ , are opened. If this is less than  $R_{max}$ , Step 8 calls ASSIGN-BEST-RELAY with the available relay nodes  $\mathcal{R}'_s$  set to  $\mathcal{R}_s$ . Otherwise, Step 10 updates the set of open relay nodes  $\mathcal{R}_s^{open}$  and Step 11 calls ASSIGN-BEST-RELAY with  $\mathcal{R}'_s$  set to  $\mathcal{R}_s^{open}$ . The algorithm assigns a primary flow to a relay node only if the achievable cooperative transmission rate is better than the primary flow's direct transmission rate, otherwise

the primary flow uses its direct transmission rate. Finally, in Step 13,  $B_p^{greedy}(i)$  is calculated for all  $i \in \mathcal{F}_p$ , which represents the best achievable transmission rates for all primary flows, based on the GORP algorithm output. In Section 5.5.1 the GORP algorithm's output is used to compute the improved primary channel utilization, which is then compared to the OFFLINE-RP-OPT relay placement solution. In Section 5.5.2, the transmission rates obtained are used as an input to the online friendly schedulers which compute the secondary VoIP capacity.

## 5.5 Simulation Model and Results

In this section we study the proposed cooperative communication system from both primary channel utilization and secondary VoIP capacity points of view. Assume the same primary network delay tolerance definition as in Section 3.6 and multi-packet data message primary VBR transmissions with a fixed message size. In order to reduce the number of system parameters and obtain better insight into the overall performance, all primary data transmission flows have identical primary traffic descriptors (message length, packet size and mean Poisson process message arrivals).

Channel fading is modelled by distance dependent path loss with lognormal shadowing [90]. The SNR of a received signal from transmitter  $u$  to receiver  $v$  is calculated as  $\eta(u, v) = \frac{P_u |h_{uv}|^2}{N_o}$ , where  $N_o$  is the Gaussian additive Noise,  $P_u$  is the transmission power and  $|h_{uv}|$  is the distance dependent log-normal distributed fading channel gain magnitude between the transmitter  $u$  and receiver  $v$ . To consider the path loss effect and shadowing, the average power gain of the distance dependent log-normal fading channel is calculated as follows:  $E[|h_{uv}|^2] = S d_{uv}^{-m}$ , where  $m$  is the path loss coefficient,  $S$  is the log-normal shadowing component and  $d_{uv}$  is the distance between the transmitter  $u$  and receiver  $v$ .

The transmit and noise power values are set as in [90]. The path loss fading exponent,  $m$ , is set to 3.5 and the time slot duration,  $\sigma$ , is taken to be 1 ms. MATLAB was used for traffic generation and for implementing the GORP algorithm and the online friendly VBR schedulers. The CVX convex optimization package [110] was used to model and solve the offline relay placement lower bound OFFLINE–RP–OPT.

### 5.5.1 Lower Bound vs. GORP Algorithm

In this section we compare the performance of the relay selection decision of the GORP algorithm to that of OFFLINE–RP–OPT. This comparison is studied under varying maximum number of open relays,  $R_{max}$ .

The results shown in Figure 5.3 are averaged over 50 simulation runs with independent randomly generated primary user layouts. For every layout,  $\mathcal{F}_p = 10$  primary users are randomly located in a circular region of radius 500 meters with the primary basestation at the centre. A grid of 16 candidate relay locations is generated around the primary basestation with a grid spacing of 200 meters. Figure 5.3 compares the achievable mean normalized primary channel utilization of OFFLINE–RP–OPT to that of the GORP algorithm. At  $R_{max} = 0$ , the primary basestation always operate in the direct transmission mode with normalized primary channel utilization of 1. The figure shows that the greater the number of open relay nodes, the lower the primary channel utilization. It also shows that the percentage of reduction in the primary channel utilization decreases as  $R_{max}$  increases. For example, at  $R_{max} = 1$ , OFFLINE–RP–OPT decreases the primary channel utilization by 40% while GORP decreases it by about 32%. However, at  $R_{max} = 2$ , both OFFLINE–RP–OPT and GORP decreases the primary channel utilization by about 20%, compared to  $R_{max} = 1$ . This percentage continues to decrease as  $R_{max}$  increases until a

saturation point is reached, i.e., opening more relay nodes does not result in any further reduction of the primary channel utilization. OFFLINE-RP-OPT reaches its saturation point at  $R_{max} = 6$  and GORP reaches its saturation point at  $R_{max} = 8$ .

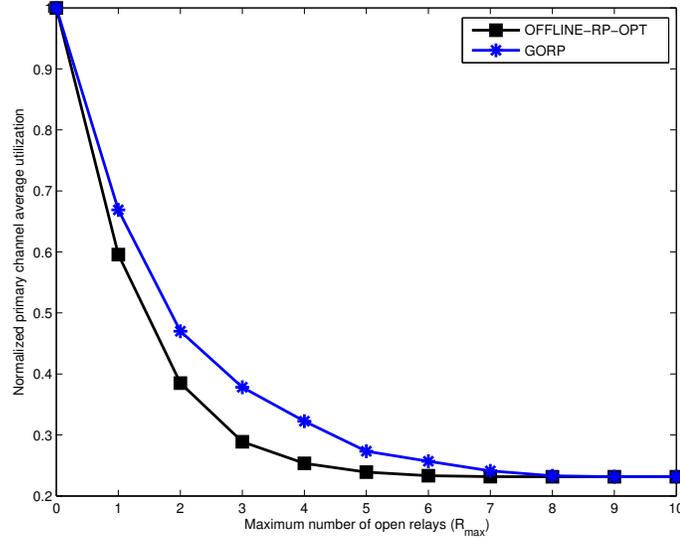


Figure 5.3: Normalized Primary Channel Utilization vs. Maximum Number of Open Relays for OFFLINE-RP-OPT Lower Bound and the GORP Algorithm.

### 5.5.2 Online Friendly VBR Schedulers with Relay Support

In this section we evaluate the secondary VoIP capacity of the proposed cooperative communication system, under varying levels of relay node cooperation and primary network delay tolerance. For a given  $R_{max}$ , the set of achievable primary flow transmission rates offered by the GORP algorithm is fed to the online friendly schedulers proposed in Section 4.4, i.e., VBR-TB and VBR-VN. We assume the same secondary network model and secondary VoIP capacity computation approach as in Section 3.6.

The primary packet size is fixed to 512 bytes for all primary flows and the primary message length is fixed to 10 packets. The Poisson process message mean arrival rate per

primary flow is set to 1/150 messages/ms and the maximum packet loss ratio for primary traffic is set to zero. The secondary VoIP traffic model, QoS parameters and codec configuration are set as in Section 4.5. A grid of 16 candidate relay locations is generated around the primary basestation with a grid spacing of 80 meters. A set of 50 independent primary user layouts is randomly generated with the same simulation parameters. For every layout,  $\mathcal{F}_p = 5$  primary users are randomly located in a circular region of radius 200 meters with the primary basestation at the centre. Each primary user layout is averaged over 10 runs, and each run is simulated over a 1000 slot (1000 ms) time window. Then the output of the GORP algorithm is computed for the 50 primary user layouts. Three representative cases are selected from our results, for further discussion below: one each with low, medium and high primary utilization.

### **Low Primary Utilization Case**

Consider the primary user layout and primary channel utilization shown in Figures 5.4 and 5.5, respectively. It can be seen in Figure 5.4 that 4 out of 5 primary users were generated very close to the primary basestation coverage boundary and would have low direct transmission rates. In this case, there is a large benefit from the relay node cooperation. It can be seen in Figure 5.5 that the percentage of reduction in the primary channel utilization when  $R_{max}$  goes from 0 to 1 is about 22%. The figure also shows that the primary channel utilization keeps decreasing as more relay nodes are opened until it is reduced by 50% when  $R_{max} \geq 4$ .

For the primary user layout shown in Figure 5.4, Figures 5.6, 5.7 and 5.8 show the secondary VoIP capacity achieved by the VBR-TB scheduler, VBR-VN scheduler in a perfect codec match case and VBR-VN scheduler in a codec mismatch case, respectively.

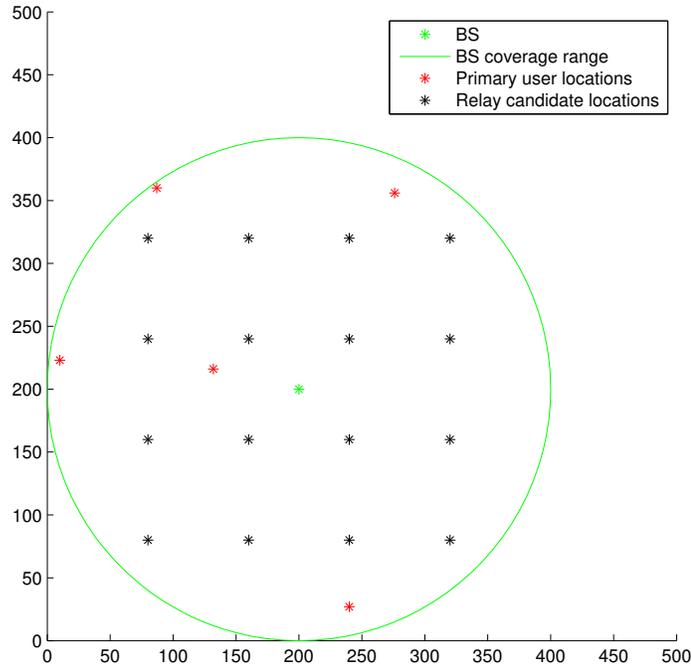


Figure 5.4: Low Primary Utilization Case: Primary User and Candidate Relay Layouts within Primary BS Coverage Range.

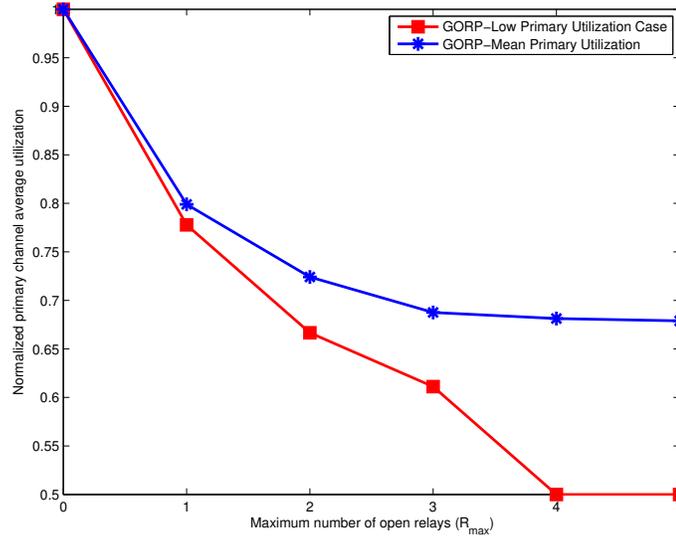


Figure 5.5: Low Primary Utilization Case: Normalized Primary Channel Utilization vs. Mean Normalized Primary Channel Utilization for the GORP Algorithm. Different values of maximum number of open relays ( $R_{max}$ ) are considered.

Comparing the three figures for a given  $D_e$ , it is shown that the more relay nodes that are opened, the more friendly the primary network is to secondary VoIP traffic. Also for given  $R_{max}$  and  $D_e$  values, the performance of the VBR-VN scheduler in the perfect codec match case, i.e., Figure 5.7, is better than its performance in the codec mismatch case (with G711-30 ms as the virtual codec used), i.e., Figure 5.8. However, both are still better than the performance of the VBR-TB scheduler, i.e., Figure 5.6, for the same traffic traces.

Additionally, Figure 5.6 shows that the VBR-TB scheduler was not able to allocate any secondary VoIP calls at  $R_{max} = 0$  and the relay node cooperation is critical in this case. The VBR-TB scheduler starts allocating secondary VoIP calls at  $R_{max} = 1$  and at  $D_e \geq 200$  ms. However, under tight packet delay constraints,  $D_e \leq 100$  ms, and at  $R_{max} \leq 3$ , the VBR-TB scheduler, i.e., Figure 5.6, achieves zero secondary VoIP capacity. The VBR-VN scheduler achieves better performance under tight delay constraints as shown in Figures 5.7 and 5.8.

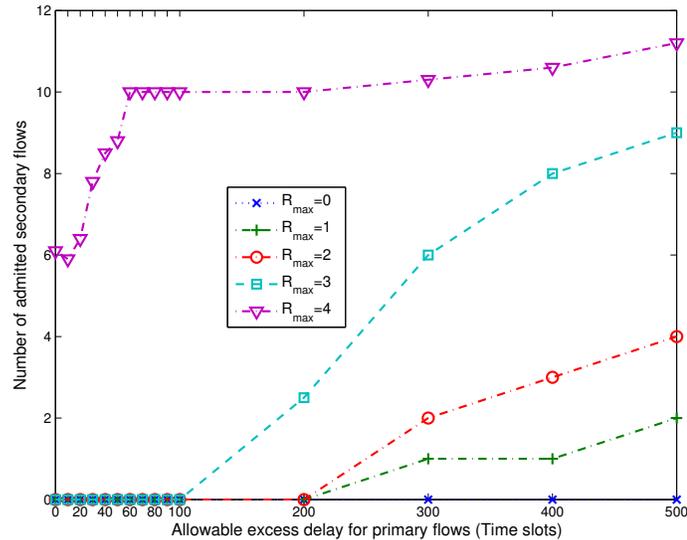


Figure 5.6: Low Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (G711-20 ms as the codec configuration).

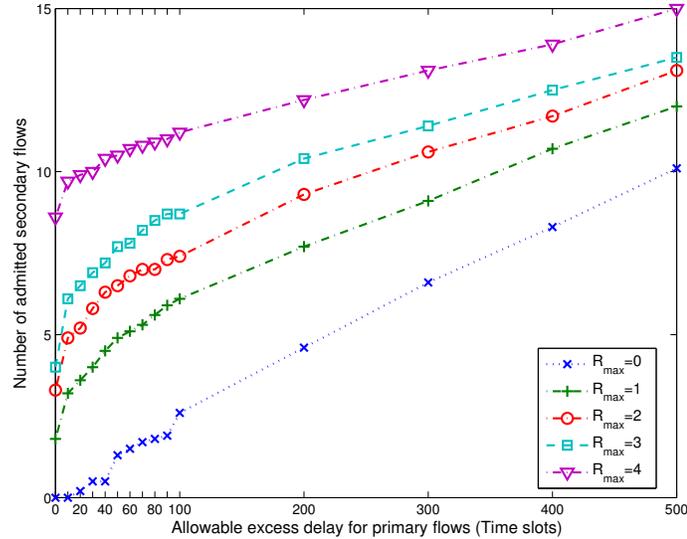


Figure 5.7: Low Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (perfect codec match case (G711-20 ms as the codec configuration)).

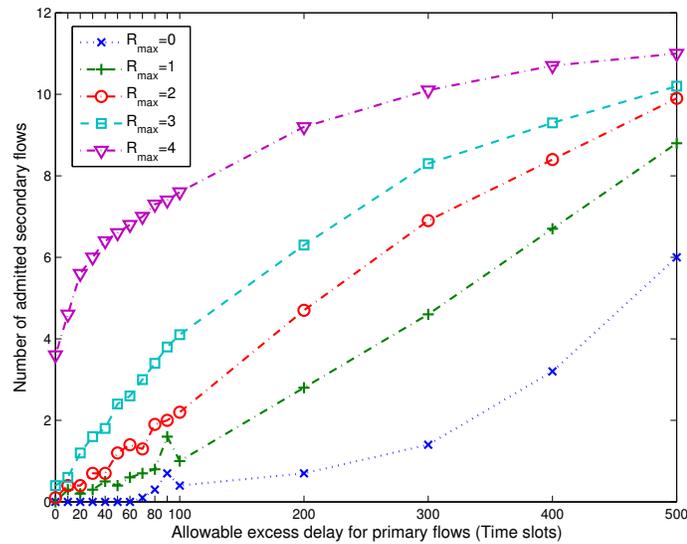


Figure 5.8: Low Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration)).

### Average Primary Utilization Case

The average primary utilization case can be analyzed in the same fashion as the previous case. The primary user layout and normalized channel utilization of the average utilization case are shown in Figures 5.9 and 5.10, respectively. The normalized primary channel utilization of this layout is the closest to the mean primary channel utilization over the 50 independent primary user layouts. We can see in Figure 5.9 that 2 out of 5 primary users would have high direct transmission rates and the remaining 3 primary users would have low direct transmission rates. In this case, 3 primary flows can greatly benefit from the relay node cooperation, while 2 primary flows can be transmitted in direct transmission mode. Figure 5.10 shows that the percentage of reduction in the primary channel utilization at  $R_{max} = 1$  is about 25%. It also shows that the primary channel utilization saturates at  $R_{max} = 2$  when the primary channel utilization is reduced by 33%.

Given this layout, Figures 5.11, 5.12 and 5.13 show the secondary VoIP capacity achieved by the VBR-TB scheduler, VBR-VN scheduler in a perfect codec match case and VBR-VN scheduler in a codec mismatch case, respectively. The performance of a given online friendly scheduler stays the same for  $R_{max} \geq 2$ . This is because the highest achievable primary flow transmission rates and the lowest achievable primary channel utilization is achieved at  $R_{max} = 2$ , as shown in Figure 5.10. Hence, for  $R_{max} \leq 2$ , the higher the number of relay nodes, the higher the secondary VoIP capacity for a given value of primary allowable excess delay  $D_e$ . The VBR-TB scheduler achieves very low secondary VoIP capacity, compared to the VBR-VN scheduler under tight delay constraints ( $D_e \leq 100$  ms).

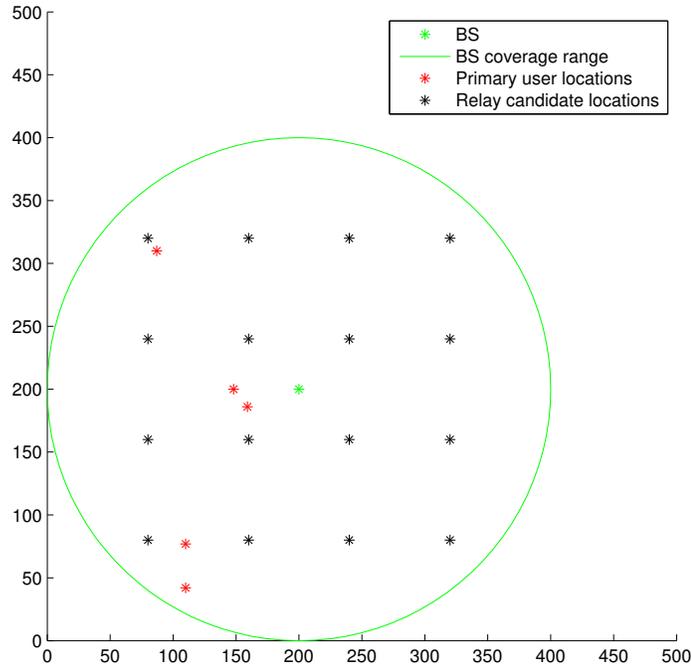


Figure 5.9: Average Primary Utilization Case: Primary User and Candidate Relay Layouts within Primary BS Coverage Range.

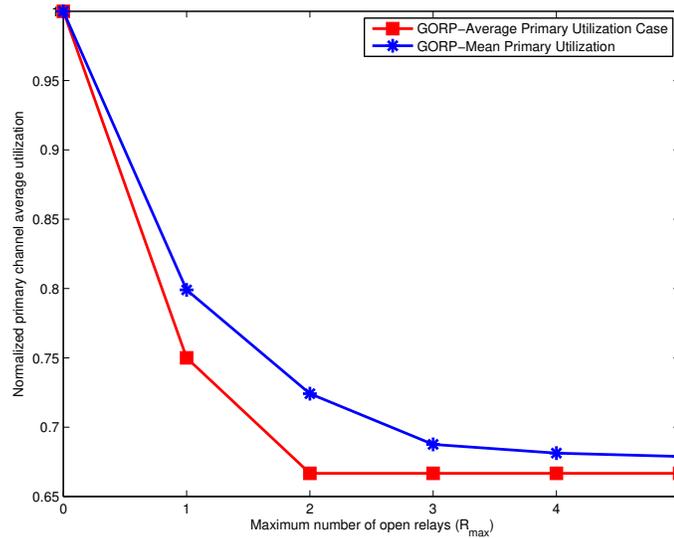


Figure 5.10: Average Primary Utilization Case: Normalized Primary Channel Utilization vs. Mean Normalized Primary Channel Utilization for the GORP Algorithm. Different values of maximum number of open relays ( $R_{max}$ ) are considered.

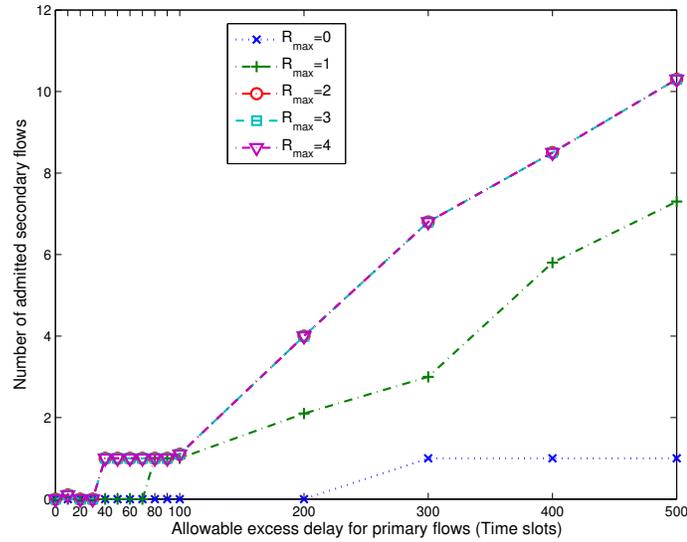


Figure 5.11: Average Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (G711-20 ms as the codec configuration).

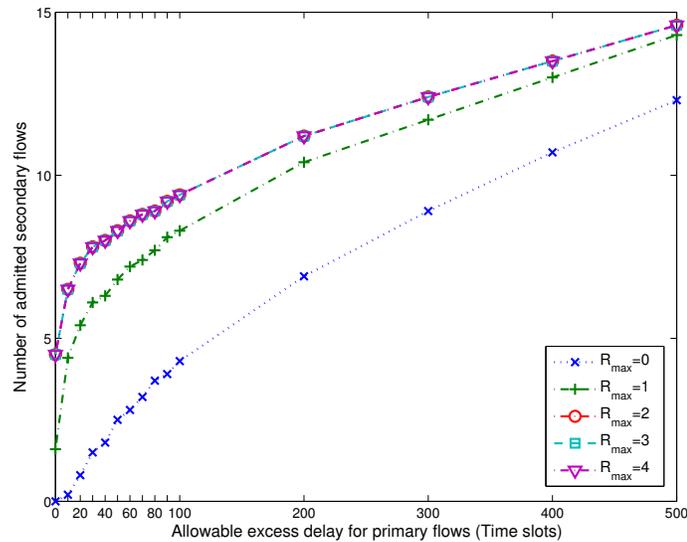


Figure 5.12: Average Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (perfect codec match case (G711-20 ms as the codec configuration)).

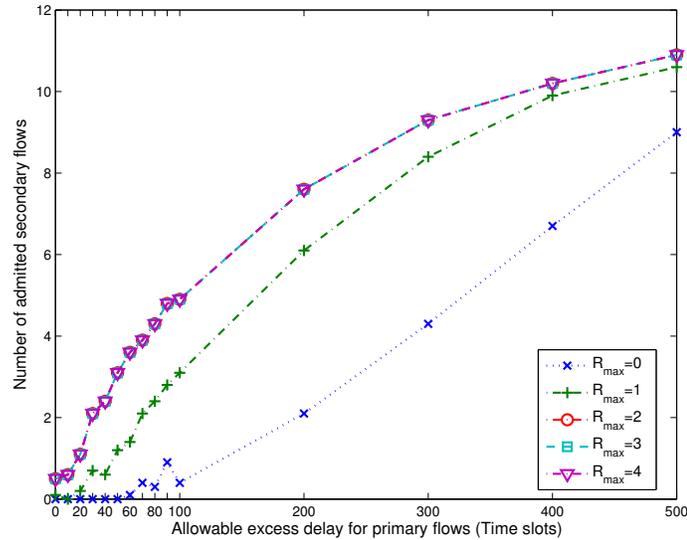


Figure 5.13: Average Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration)).

### High Primary Utilization Case

The primary user layout and normalized channel utilization of the high utilization case are shown in Figures 5.14 and 5.15, respectively. Given this layout, the direct primary flow transmission rates are typically better than their best achievable cooperative rates. This is obvious in Figure 5.15, where the percentage of reduction in the primary channel utilization at  $R_{max} = 1$  is only about 11%. The figure also shows that the primary channel utilization saturates at  $R_{max} = 1$ , i.e., opening more than one relay node does not result in any further reduction of the primary channel utilization. Hence, most of the primary flows are transmitted in the direct transmission mode. In this case, the cooperation with relay nodes has a minimal effect on the primary channel utilization and secondary VoIP capacity. This is shown in Figures 5.16, 5.17 and 5.18, where the performance of a given online

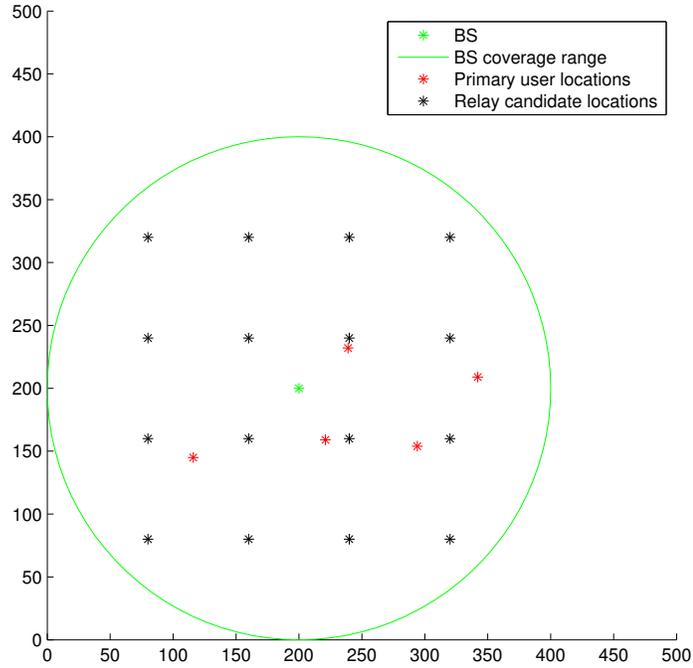


Figure 5.14: High Primary Utilization Case: Primary User and Candidate Relay Layouts within Primary BS Coverage Range.

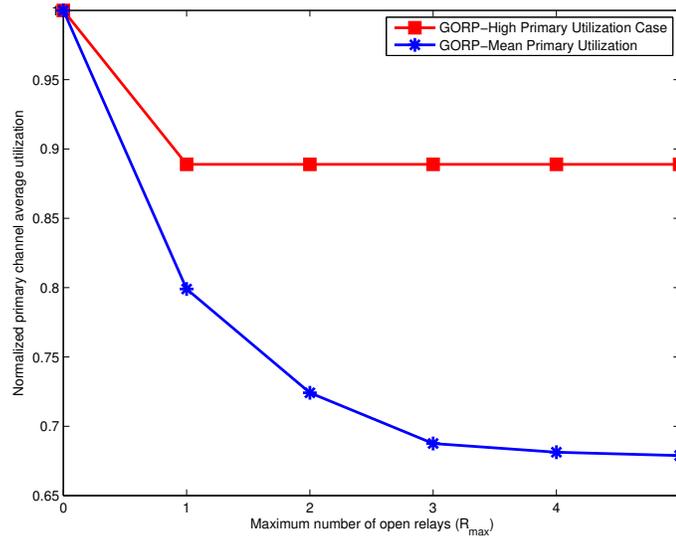


Figure 5.15: High Primary Utilization Case: Normalized Primary Channel Utilization vs. Mean Normalized Primary Channel Utilization for the GORP Algorithm. Different values of maximum number of open relays ( $R_{max}$ ) are considered.

friendly scheduler stays the same for  $R_{max} \geq 1$ . A minimal improvement in the secondary VoIP capacity is achieved by the VBR-VN algorithm, at  $R_{max} = 1$ , because the primary channel utilization is only improved by 11%, as shown in Figure 5.15.

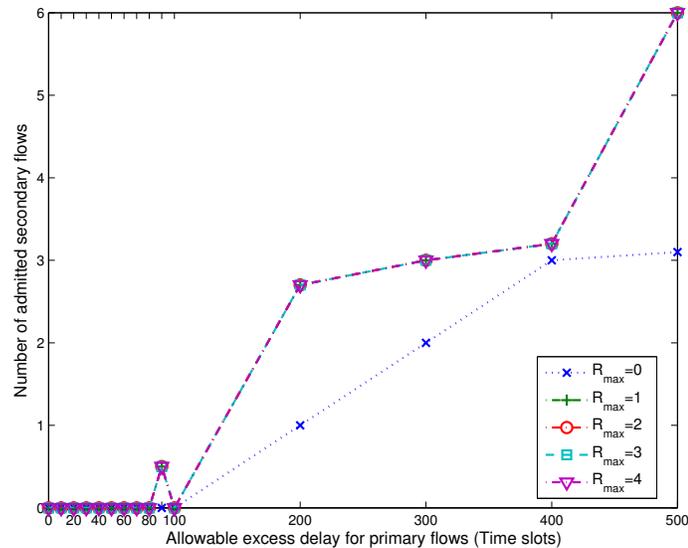


Figure 5.16: High Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-TB Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (G711-20 ms as the codec configuration).

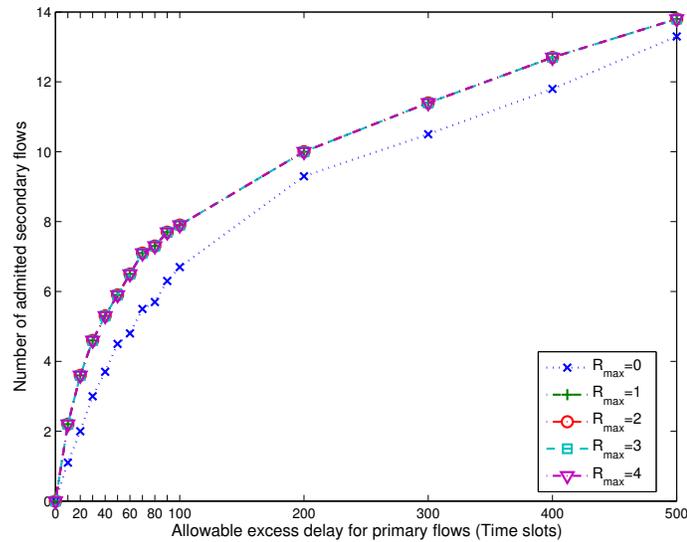


Figure 5.17: High Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (perfect codec match case (G711-20 ms as the codec configuration)).

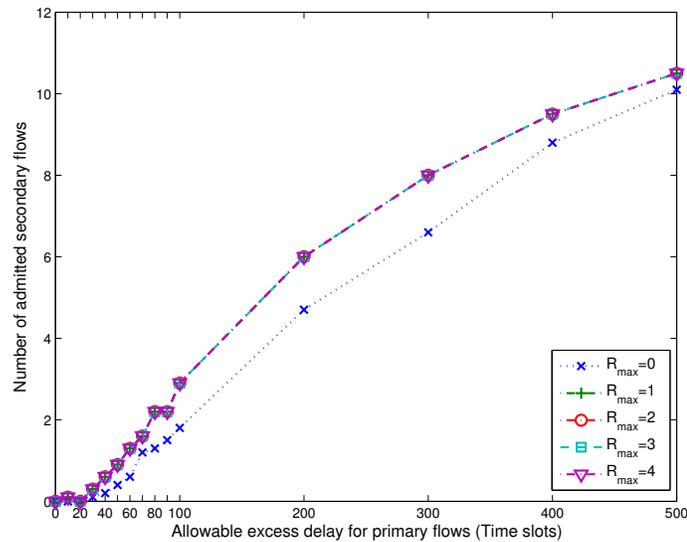


Figure 5.18: High Primary Utilization Case: Secondary VoIP Capacity vs. Primary Delay Tolerance for the VBR-VN Scheduler. Different values of maximum number of open relays ( $R_{max}$ ) are considered (codec mismatch case (G711-20 ms as the real codec configuration and G711-30 ms as the virtual codec configuration)).

## 5.6 Conclusions

In this chapter we have proposed a communication framework where the primary basestation cooperate with fixed relay nodes to reduce the primary channel utilization and increase the capacity available for sharing by secondary users. The relay nodes forward user traffic from the primary basestation using the DF relaying protocol in the time domain, which, in turn, increases the primary traffic transmission rates. The previously proposed online friendly schedulers were then used to shape the enhanced spectrum hole pattern to be friendly to potential secondary VoIP calls.

An offline relay placement ILP was first presented, whose solution is a lower bound on the primary channel utilization. The lower bound solution is the set of optimal relay locations that minimize the number of time slots occupied by primary traffic.

An offline relay placement (GORP) algorithm was then introduced, which provides a relay selection decision by making greedy mappings of primary flows to relay nodes. The GORP algorithm was compared against the lower bound for different numbers of available relay nodes and was found to perform well compared to the bound. Fifty independent randomly located primary user layouts were generated. Three representative primary user layouts were then selected and discussed in detail, based on different levels of achievable primary channel utilization. The results showed that the effect of the proposed cooperative framework on the secondary VoIP capacity depends on the primary user layout and the number of cooperative relay nodes. Increasing the number of open relay nodes directly increase the secondary VoIP capacity achieved by the online friendly schedulers. This was especially clear in the low and average primary channel utilization cases and particularly with the VBR-TB scheduler under tight primary delay requirements. However, this increase is more significant in the low utilization case compared to the average utilization

one. In the high primary channel utilization case, increasing the number of the cooperative relay nodes provided a minimal effect on the primary channel utilization and secondary VoIP capacity. It can be concluded that the proposed cooperative framework is effective when a significant number of the primary users are located far enough from the primary basestation. In this case, the cooperation between the primary basestation and relay nodes can significantly increase the secondary VoIP friendliness by expanding the secondary VoIP capacity achieved by the online friendly schedulers. As in the results presented in Chapter 4, the secondary VoIP capacity achieved by the VBR-VN scheduler surpassed that of the VBR-TB scheduler, especially under tight primary delay requirements. The performance of the VBR-VN scheduler largely depends on the difference between the virtual VoIP codec assumed by the VBR-VN scheduler and the real VoIP codec used by the secondary users. The VBR-VN performance in the VoIP codec match case was always better than its performance in any VoIP codec mismatch case, for the same simulation parameters.

# Chapter 6

## Conclusions and Future Work

Cognitive radio networks provide efficient solutions to the spectrum underutilization problem caused by fixed spectrum assignment policies. Using flexible spectrum management techniques, CRNs allow unlicensed (secondary) users to share the spectrum with licensed (primary) users in a variety of ways that best protect the QoS requirements of primary users. With the growing demand for wireless multimedia services, supporting real-time services over CRNs is becoming increasingly important.

In conventional CRNs, the primary network usually remains unchanged. In some cases however, the primary network operator may wish to accommodate secondary user access to their radio spectrum. An efficient QoS-aware primary scheduler can play an important role in supporting QoS over the secondary network, by operating in a way that is friendly to potential real-time secondary services.

This thesis considers the problem of supporting VoIP over CRNs, but with emphasis on secondary network “friendliness” from the primary network point of view. Friendliness is measured by the secondary VoIP capacity, defined as the number of VoIP connections that can be supported subject to typical QoS constraints.

Secondary friendliness was first considered from the primary basestation scheduling point of view by maximizing the secondary VoIP capacity based on relocating the spectrum holes. The air interface types studied were constant bit rate (CBR) and variable bit rate (VBR) in Chapters 3 and 4, respectively. In Chapter 5, secondary friendliness was studied from the primary infrastructure deployment point of view by using cooperative primary transmission to increase the secondary VoIP capacity.

In Chapter 3, the CBR case was examined when the primary basestation is transmitting data packets subject to delay constraints and the secondary users are transmitting real-time VoIP traffic. In order to provide an optimum upper bound on friendliness, an integer linear program (ILP) was formulated, which was then solved using a minimum cost flow graph construction in time complexity that is polynomial in the number of time slots. Two online friendly traffic schedulers were then proposed. The first (TB scheduler) is based on shaping the primary traffic, which alters the spectrum holes available to secondary traffic, and the second (VN scheduler) is based on scheduling both the real primary traffic and virtual secondary VoIP traffic with priority given to the primary traffic. The results showed that the higher the delay tolerance of the primary traffic, the more secondary friendly it can be. The proposed schedulers perform significantly better than a conventional non-friendly first come first serve (FCFS) scheduler especially when primary traffic consists of bursty message based arrivals with a high delay tolerance. The proposed schedulers also performed well compared to the optimum bound. Results showed that the average primary packet arrival rate can be considered a good candidate value for the minimum token rate threshold, which is set to improve the scheduling feasibility. It was shown that the performance of the VN algorithm largely depends on the choice of the codec configuration of the virtual and real VoIP traffic. In the case of perfect codec match, the VN scheduler performs better

than the TB scheduler. In the case of codec mismatch however, the relative performance of the two schedulers depends on the codec configuration. It was found that the TB scheduler performance is only dependent on the primary traffic, however it is completely non-friendly (zero secondary VoIP capacity) under very tight primary delay constraints when compared to the VN scheduler. Experimenting with different arrival processes for the primary traffic, our results showed that the heavier the tail of the distribution, the lower the secondary VoIP capacity that can be achieved by the proposed schedulers. Simulation results also showed that varying the QoS requirements of the secondary VoIP traffic directly affect the achievable secondary VoIP capacity. In particular, the jitter requirement of the secondary VoIP call significantly affects the friendliness level of the primary network.

In Chapter 4, the VBR interface was assumed for the primary transmissions. This chapter considered multiple primary users, each receiving data packets subject to delay constraints from the primary basestation at different transmission bit rates. An offline VBR scheduler, OFFLINE-VBR-OPT, was developed to maximize friendliness. Two online friendly VBR schedulers were then presented, VBR-TB and VBR-VN, based on the TB and VN friendly schedulers. The proposed VBR online schedulers performed well compared to the offline scheduler. Considering primary data flows with identical traffic descriptors, the effect of changing the mean primary message arrival rate and the primary message length was studied under varying primary delay tolerance. The performance of the VBR-VN scheduler in the perfect codec match case was generally better than both its performance in the codec mismatch case and the performance of the VBR-TB scheduler, for the same traffic traces. Moreover, in the representative codec mismatch case, the VBR-VN scheduler achieved performance that is better than that of the VBR-TB scheduler, under relatively tight primary delay requirements. However, under relaxed primary

delay requirements, the VBR-TB scheduler performed better than the VBR-VN scheduler in the representative codec mismatch case with some restrictions on the message length and mean message arrival rate values. Hence, the VBR-TB friendly scheduler is not the scheduler of choice under tight primary delay requirements. Primary message lengths that are larger than the secondary VoIP call jitter requirement tended to block the channel for longer periods, making it more difficult for VoIP calls to coexist. We also found that primary flows transmitted at lower bit rates have a greater impact on increasing the primary channel utilization as a result of increasing the primary traffic load per flow, compared to primary flows transmitted at higher bit rates.

In Chapter 5, a cooperative communication framework was proposed. Based on a decode-and-forward (DF) relaying technique, the primary basestation cooperates with fixed relay nodes to reduce the primary channel utilization and increase the spectrum holes available for potential secondary users. An offline relay placement ILP was presented, whose solution is a lower bound on the primary channel utilization. A greedy offline relay placement (GORP) algorithm was then introduced, which provides a relay selection decision by mapping primary data flows to relay nodes. The GORP algorithm was compared against the lower bound for different numbers of relay nodes and found to perform well. The VBR-TB and VBR-VN friendly schedulers were then used to shape the enhanced spectrum hole pattern to be friendly to potential secondary VoIP calls. Three representative levels of achievable primary channel utilization were selected and discussed in detail. The results showed that the effect of increasing the number of open relay nodes directly increase the secondary VoIP capacity achieved by the online friendly schedulers. This was especially clear in the low and average primary channel utilization cases and particularly

with the VBR-TB scheduler under tight primary delay requirements. However, this increase is more significant in the low utilization case compared to the average utilization case. In the high primary channel utilization case, increasing the number of the cooperative relay nodes provided a minimal effect on the primary channel utilization and secondary VoIP capacity. Hence, the proposed cooperative framework is effective when a significant number of the primary users are located far enough from the primary basestation. In this case, the cooperation between the primary basestation and relay nodes can significantly increase the secondary VoIP friendliness by expanding the secondary VoIP capacity achieved by the online friendly schedulers.

In the future, the work in this thesis can be extended by implementing a spectrum pricing scheme by the primary network, where the secondary network is charged for primary network friendliness. This can compensate for the degradation in the primary network performance as a consequence of secondary user friendliness. In our case for example, primary messages experience longer delays without violating their deadlines. Spectrum pricing models have been previously proposed for spectrum sharing in CRNs. In [111] for example, game theory is used to achieve an optimal solution between the primary and secondary users. Similar approaches could be applied to our work, considering the friendliness aspect of the problem. Moreover, the single channel assumption in our work can be relaxed by considering multiple channels where secondary users can opportunistically share spectrum holes across several channels. This would require designing a multichannel friendly primary scheduler that creates friendly spectrum hole patterns across multiple channels. The time invariant channel assumption can also be relaxed by considering a time variant channel model, which would require relevant modifications to the proposed online friendly schedulers.

Our simulation results considered primary traffic flows with identical traffic descriptors. However, it would be interesting to study secondary friendliness of a primary network with different classes of traffic, i.e., different levels of QoS requirements. It would be also interesting to study secondary friendliness when the primary network is transmitting VoIP traffic. Underlay spectrum sharing could also be considered instead of interweave spectrum sharing, to allow opportunistic use of the underused spectrum holes as well as unused spectrum holes. In an underlay spectrum sharing paradigm, a friendly primary scheduler would adjust the friendliness of the primary network by tuning the amount of interference it can bear, which could be done on a periodic basis to accommodate secondary VoIP services.

# Appendix A

## Secondary VoIP Friendliness Using an STDMA Message Delay Model

In this appendix an STDMA message delay model is used to approximate secondary user friendliness. Results are presented that compare the model to the online schedulers proposed in Section 3.5.

In [7], a message delay analysis was presented for a synchronous time division multiple access (STDMA) system for multiple consecutive outputs and multi-packet messages. In this model, time is divided into frames, each consisting of  $T_F$  time slots. Capacity in the form of  $b \geq 1$  time slots per frame is allocated to each traffic source sharing the channel, and one fixed-length packet is transmitted per time slot. For a particular traffic source, Reference [7] gives an exact message delay analysis for generic arrival and message length distributions. An infinite capacity buffer is assumed with FCFS queueing at the traffic source. The queue length distribution is derived using an imbedded Markov chain as seen by a newly arrived tagged message. For an arbitrary message arrival process and arbitrary message length distribution, the probability generating function,  $X(z)$ , of the number of

pending packets at the traffic source queue,  $\tilde{x}$  is given by

$$X(z) = \frac{[A(M(z)) - 1](1 - z)(b - \mu\lambda_F)}{[M(z) - 1]\lambda_F[A(M(z)) - z^b]} \prod_{\tau=1}^{b-1} \frac{z - z_\tau}{1 - z_\tau} \quad (\text{A.1})$$

where  $A(z)$  is the probability generating function of the number of message arrivals per frame,  $\tilde{a}$ , with mean  $\lambda_F$  (messages per frame).  $M(z)$  is the probability generating function of the number of packets per message,  $\tilde{m}$ , with mean  $\mu$ , and  $b$  is the number of time slots per frame assigned to the traffic source. Reference [7] also obtains the message delay analysis based on the queueing length result. As shown in Figure A.1, the total message delay is divided into three components, i.e.,

$$\tilde{d} = \tilde{d}_1 + \tilde{d}_2 + \tilde{d}_3 \quad (\text{A.2})$$

where the random variable  $\tilde{d}_1$  is the synchronization delay due to the ‘‘Please Wait’’ strategy [112] and the average value of  $\tilde{d}_1$  is half of a frame duration. The random variable  $\tilde{d}_2$  is the number of frames needed to transmit the messages pending before the tagged message, which is calculated as

$$\tilde{d}_2 = \left\lceil \frac{\tilde{x} + \tilde{m} - 1}{b} \right\rceil \quad (\text{A.3})$$

Finally, the random variable  $\tilde{d}_3$  is the number of time slots needed in the final transmission frame to complete the transmission of the tagged message, which is given by

$$\tilde{d}_3 = 1 + [(\tilde{x} + \tilde{m} - 1) \bmod b] \quad (\text{A.4})$$

The analysis presented above can be applied to arbitrary message arrival process and arbitrary message length distributions. We show how this can be used to approximate the

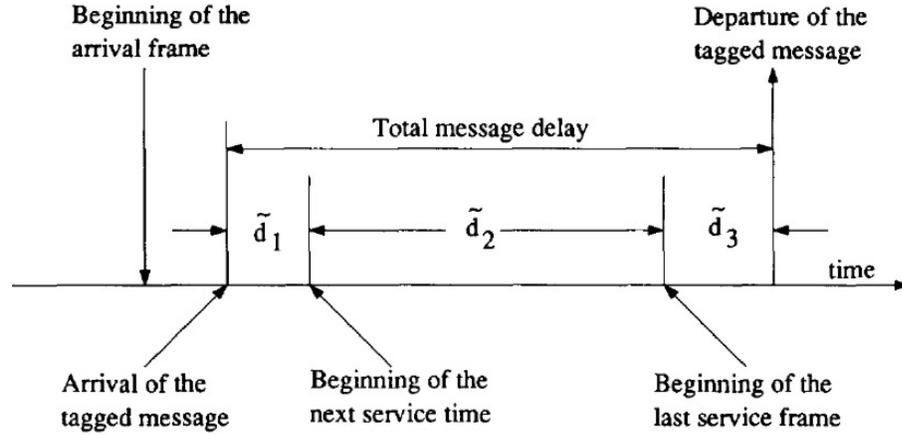


Figure A.1: Components of STDMA Message Delay © [1990] IEEE [7]

secondary user VoIP capacity when there is friendly scheduling at the primary basestation.

## A.1 Secondary VoIP Capacity Estimation using STDMA Delay Analysis

The primary traffic source is assigned  $b$  time slots per frame and the rest of the frame, i.e.,  $(T_F - b)$  time slots, is assigned to secondary traffic, one slot per secondary flow.  $T_F$  represents the packetization period and the jitter requirement of the constant bit rate secondary VoIP flows. The online schedulers use the allowable excess delay to represent the primary delay tolerance and set the primary packet deadlines. It has been noted in our results, that the excess delay typically dominates the primary packet delay over a wide range of parameter values. For this reason, in the STDMA analysis, we use the excess delay as the primary delay requirement.

The STDMA analysis [7] applies to arbitrary arrival distributions. In our case, for example, we model the Poisson process primary packet arrivals using a geometric message

arrival process with a mean of  $\lambda$  (messages per slot) and with fixed message lengths,  $c$ . Substituting these into Equation (A.1), we obtain

$$X(z) = \frac{(1-z)(b-\lambda T_F c)}{\lambda T_F z^{(b+c)} - (1+\lambda T_F)z^b + 1} \prod_{\tau=1}^{b-1} \frac{z-z_\tau}{1-z_\tau} \quad (\text{A.5})$$

The denominator of  $X(z)$  in Equation (A.5) has  $(b+c)$  zeros,  $c$  of which lie outside the unit circle,  $|z| > 1$ , in the complex plane. One of these zeros is 1, and the remaining  $(b-1)$  zeros lie inside the unit circle,  $|z| \leq 1$ . Using this result, Equation (A.5) can be simplified as

$$X(z) = \prod_{\tau=1}^c \frac{(1-z_\tau)}{(z-z_\tau)} \quad (\text{A.6})$$

where  $z_1, z_2, \dots, z_c$  are the zeros of the denominator of  $X(z)$  in Equation (A.5) for which  $|z| > 1$  is true. The probability mass function of  $\tilde{x}$  is then derived, using the inverse  $z$ -transform, as

$$\Pr[\tilde{x} = n] = (-1)^{(c+1)} \prod_{\tau=1}^c (z_\tau - 1) \sum_{i=1}^c \frac{z_i^{(-n-1)}}{\prod_{j=1, j \neq i}^c (z_i - z_j)} \quad (\text{A.7})$$

and for a constant message length,  $c$ , the probability density function of  $\tilde{m}$  is

$$\Pr[\tilde{m} = c] = 1 \quad (\text{A.8})$$

Equations (A.7) and (A.8) are then used to derive the joint probability density function of the total message delay  $\tilde{d}_2 + \tilde{d}_3$ . This is done by substituting in Equations (A.3) and (A.4) with all values for  $\tilde{x}$  to obtain all possible values of  $\tilde{d}_2 + \tilde{d}_3$  and their corresponding probabilities from Equations (A.7) and (A.8). The inverted cumulative density function of  $\tilde{d}_2 + \tilde{d}_3$  is calculated from its probability density function. This, in turn, is used to

determine the drop percentage of the primary traffic for a given primary delay requirement. For geometric message arrivals and fixed message lengths, for example, this is done as follows. First, we obtain the minimum value of  $b$  that satisfies the maximum allowable drop percentage for the primary traffic based on the inverted cumulative density function of the primary message total delay  $\tilde{d}_2 + \tilde{d}_3$ . This is done by trying all values of  $b$ , in descending order, that satisfy  $\lambda T_{FC} < b \leq T_F$  and checking if the estimated primary drop percent corresponding to the given primary delay requirement on the message delay CDF is less than the maximum allowable primary drop percent,  $\epsilon$ . The minimum value of  $b$ ,  $b_{min}$ , that satisfies the conditions above, gives the secondary VoIP capacity that corresponds to  $(T_F - b_{min})$ .

To estimate the secondary VoIP capacity using the STDMA analysis described above, the primary traffic descriptors ( $\lambda$ ,  $c$ ), and the primary traffic QoS constraints (delay requirement and  $\epsilon$ ) are required. The STDMA analysis also requires the packetization period for the VoIP Codec used for the secondary flows. Since the STDMA analysis assigns time slots to every secondary flow every  $T_F$  time slots, the jitter and delay measurements of the secondary packets are forced to be equal to  $T_F$  and the packet loss ratio of the secondary traffic is forced to zero.

Figures A.2 and A.3 show the estimated secondary VoIP capacity using the STDMA analysis for varying primary delay tolerance compared to the proposed online friendly scheduling algorithms (TB and VN). The mean of the geometric primary message arrival process,  $\lambda$ , is set to 1/100 messages/ms. The primary message length,  $c$ , is fixed and set to 30 packets/message, the primary packet loss percent,  $\epsilon$ , is set to 0.5%. The secondary VoIP codec is set to G711 with the packetization period set to 20 ms and 30 ms in Figures A.2 and A.3, respectively. For these secondary VoIP codec configurations, the channel slot

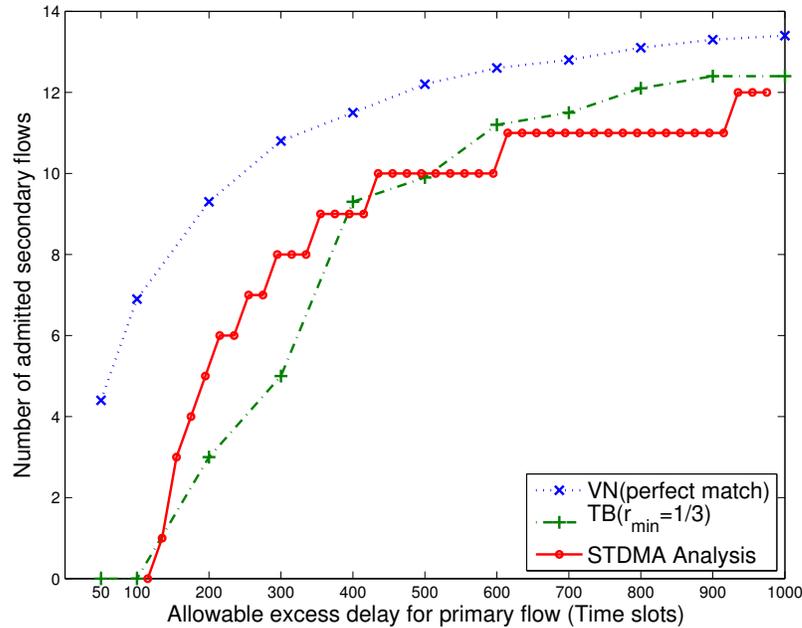


Figure A.2: Secondary VoIP Capacity vs. Primary Delay Tolerance. The STDMA analysis is compared to the TB and VN schedulers (G711-20 ms codec configuration).

size can carry only one secondary packet. The VN algorithm operates in the perfect codec match mode and the TB algorithm has the  $r_{min}$  value set to the mean packet arrival rate.

The STDMA analysis shows a monotonic behaviour similar to that achieved by the online algorithms. However, the online algorithms, especially VN, can achieve better secondary VoIP capacity compared to the STDMA analysis. This is due to the fact that the online algorithms are fed with actual traffic traces for the primary and secondary traffic. Although the simulations are run for long simulation time windows and results are averaged over 10 runs, the randomness in the geometric arrival process causes some differences, especially for primary excess delay  $> 400$  ms, between the online algorithms and the theoretical STDMA analysis that are dependent on the traffic descriptors. However, the STDMA analysis gives a good correspondence with the online algorithms (especially the TB scheduler), which gives good evidence that the simulation models are operating

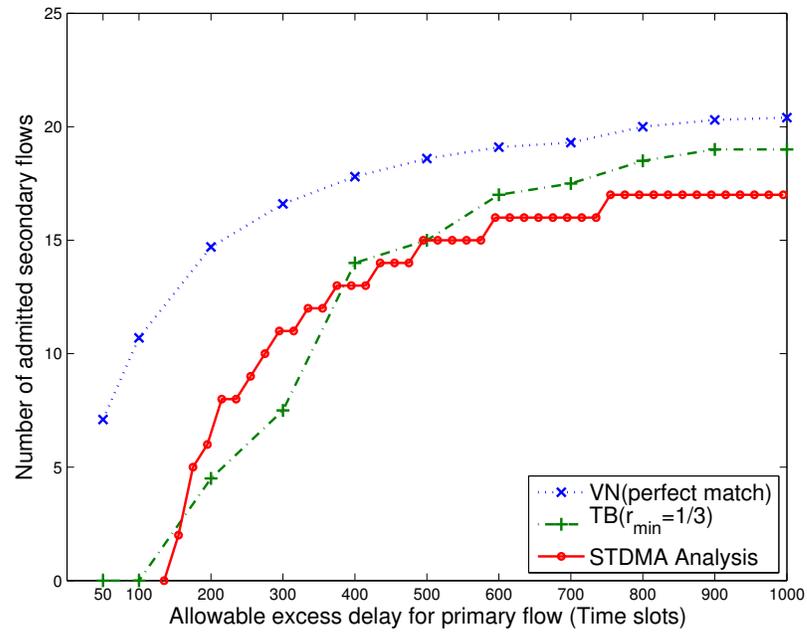


Figure A.3: Secondary VoIP Capacity vs. Primary Delay Tolerance. The STDMA analysis is compared to the TB and VN schedulers (G711-30 ms codec configuration).

correctly.

# Bibliography

- [1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, “Next Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey,” *Computer networks*, vol. 50, no. 13, pp. 2127–2159, 2006.
- [2] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, “CRAHNs: Cognitive Radio Ad Hoc Networks,” *AD hoc networks*, vol. 7, no. 5, pp. 810–836, 2009.
- [3] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, “A Survey on Spectrum Management in Cognitive Radio Networks,” *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, 2008.
- [4] P. Wang, D. Niyato, and H. Jiang, “Voice-Service Capacity Analysis for Cognitive Radio Networks,” *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1779–1790, 2010.
- [5] O. Simeone, I. Stanojev, S. Savazzi, Y. Bar-Ness, U. Spagnolini, and R. Pickholtz, “Spectrum Leasing to Cooperating Secondary Ad Hoc Networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 1, pp. 203–213, 2008.

- [6] C. Zhai and W. Zhang, "Adaptive Spectrum Leasing with Secondary User Scheduling in Cognitive Radio Networks," *Wireless Communications, IEEE Transactions on*, vol. 12, no. 7, pp. 3388–3398, 2013.
- [7] H.-W. Lee and L. Liang, "A Generalized Analysis of Message Delay in STDMA," *Computer Networks and ISDN systems*, vol. 19, no. 1, pp. 11–24, 1990.
- [8] FCC, "Notice of Proposed Rule Making and Order," FCC Document ET Docket No 03-222, December 2003.
- [9] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *Selected Areas in Communications, IEEE Journal on*, vol. 23, no. 2, pp. 201–220, 2005.
- [10] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa, "Breaking Spectrum Gridlock with Cognitive Radios: An Information Theoretic Perspective," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 894–914, 2009.
- [11] J. Mitola Iii, "Cognitive Radio for Flexible Mobile Multimedia Communications," in *1999 IEEE International Workshop on Mobile Multimedia Communications, 1999 (MoMuC'99)*. IEEE, 1999, pp. 3–10.
- [12] C. V. N. Index, "Cisco Visual Network Index: Global Mobile Traffic Forecast Update," *Cisco white paper*, 2013.
- [13] L. Ericsson, "More than 50 Billion Connected Devices," *White Paper*, 2011.

- [14] C.-X. Wang, F. Haider, X. Gao, X.-H. You, Y. Yang, D. Yuan, H. M. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir, "Cellular Architecture and Key Technologies for 5G Wireless Communication Networks," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 122–130, 2014.
- [15] X. Hong, J. Wang, C.-X. Wang, and J. Shi, "Cognitive Radio in 5G: A Perspective on Energy-Spectral Efficiency Trade-off," *IEEE Communications Magazine*, vol. 52, no. 7, pp. 46–53, 2014.
- [16] K. Pelechrinis, P. Krishnamurthy, M. Weiss, and T. Znati, "Cognitive Radio Networks: Realistic or Not?" *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 2, pp. 44–51, 2013.
- [17] C. Cormio and K. Chowdhury, "A Survey on MAC Protocols for Cognitive Radio Networks," *Elsevier Journal on Ad Hoc Networks*, vol. 7, no. 7, pp. 1315–1329, 2009.
- [18] L. Hu, V. B. Iversen, and L. Dittmann, "Survey of PHY and LINK Layer Functions of Cognitive Radio Networks for Opportunistic Spectrum Sharing," in *Communications and Networking in China*. Springer, 2009, pp. 10–24.
- [19] H. Wang, H. Qin, and L. Zhu, "A Survey on MAC Protocols for Opportunistic Spectrum Access in Cognitive Radio Networks," in *Proc. IEEE 2008 Int. Conf. Comput. Sci. Software Eng.*, vol. 1, 2008, pp. 214–218.
- [20] P. Pawelczak, R. V. Prasad, L. Xia, and I. G. Niemegeers, "Cognitive Radio Emergency Networks-Requirements and Design," in *New Frontiers in Dynamic Spectrum*

*Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on.* Ieee, 2005, pp. 601–606.

- [21] L. Ma, X. Han, and C. Shen, “Dynamic Open Spectrum Sharing MAC Protocol for Wireless Ad hoc Networks,” in *Proc. IEEE DySPAN’05*, Nov. 2005, pp. 203–213.
- [22] N. Choi, M. Patel, and S. Venkatesan, “A Full Duplex Multi-Channel MAC Protocol for Multi-Hop Cognitive Radio Networks,” in *Cognitive Radio Oriented Wireless Networks and Communications, 2006. 1st International Conference on.* IEEE, 2006, pp. 1–5.
- [23] C. Hsu, D. Weit, and C. Kuo, “A Cognitive MAC Protocol using Statistical Channel Allocation for Wireless Ad-hoc Networks,” in *Proc. IEEE WCNC’07*, March 2007, pp. 105–110.
- [24] S. Lien, C. Tseng, and K. Chen, “Carrier Sensing Based Multiple Access Protocols for Cognitive Radio Networks,” in *Proc. IEEE ICC’08*, May 2008, pp. 3208–3214.
- [25] B. Hamdaoui and K. G. Shin, “OS-MAC: An Efficient MAC Protocol for Spectrum-Agile Wireless Networks,” *Mobile Computing, IEEE Transactions on*, vol. 7, no. 8, pp. 915–930, 2008.
- [26] J. Jia, Q. Zhang, and X. Shen, “HC-MAC: A Hardware-Constrained Cognitive MAC for Efficient Spectrum Management,” *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 106–117, 2008.
- [27] C. Cordeiro and K. Challapali, “C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Networks,” in *Proc. IEEE DySPAN’07*, April 2007, pp. 147–157.

- [28] Y. Kondareddy and P. Agrawal, "Synchronized MAC Protocol for Multi-hop Cognitive Radio Networks," in *Proc. IEEE ICC'08*, May 2008, pp. 3198–3202.
- [29] Y.-C. Liang, Y. Zeng, E. C. Peh, and A. T. Hoang, "Sensing-Throughput Tradeoff for Cognitive Radio Networks," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 4, pp. 1326–1337, 2008.
- [30] H. Jiang, L. Lai, R. Fan, and H. Poor, "Optimal selection of channel sensing order in cognitive radio," *IEEE Transactions on Wireless Communications*, vol. 8, no. 1, pp. 297–307, 2009.
- [31] R. Fan and H. Jiang, "Channel Sensing-Order Setting in Cognitive Radio Networks: A Two-User Case," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 4997–5008, 2009.
- [32] —, "Optimal Multi-Channel Cooperative Sensing in Cognitive Radio Networks," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 3, pp. 1128–1138, 2010.
- [33] IEEE 802.22 Working Group on Wireless Regional Area Networks. [Online]. Available: <http://www.ieee802.org/22/>
- [34] R. Urgaonkar and M. Neely, "Opportunistic Scheduling with Reliability Guarantees in Cognitive Radio Networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 6, pp. 766–777, 2009.
- [35] B. Wang and D. Zhao, "Scheduling for Long Term Proportional Fairness in a Cognitive Wireless Network with Spectrum Underlay," *IEEE Transactions on Wireless Communications*, vol. 9, no. 3, pp. 1150–1158, 2010.

- [36] A. Hoang, Y. Liang, D. Wong, Y. Zeng, and R. Zhang, "Opportunistic Spectrum Access for Energy-Constrained Cognitive Radios," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1206–1211, 2009.
- [37] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized Cognitive MAC for Opportunistic Spectrum Access in Ad hoc Networks: A POMDP Framework," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 3, pp. 589–60, 2007.
- [38] H. Su and X. Zhang, "Cross-Layer Based Opportunistic MAC Protocols for QoS Provisionings over Cognitive Radio Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 118–129, 2008.
- [39] L. Le and E. Hossain, "OSA-MAC: A MAC Protocol for Opportunistic Spectrum Access in Cognitive Radio Networks," in *Proc. IEEE WCNC'08*, 2008, pp. 1426–1430.
- [40] H. Song and X. Lin, "A Group Based MAC Protocol for QoS Provisioning in Cognitive Radio Networks," in *Proc. IEEE ICCS'08*, Nov. 2008, pp. 1489–1493.
- [41] Y. Wu and D. Tsang, "Distributed Power Allocation Algorithm for Spectrum Sharing Cognitive Radio Networks with QoS Guarantee," in *Proc. IEEE INFOCOM'09*, April 2009, pp. 981–989.
- [42] C. Zou and C. Chigan, "A Game Theoretic DSA-Driven MAC Framework for Cognitive Radio Networks," in *Proc. IEEE ICC'08*, May 2008, pp. 4165–4169.
- [43] Y. Xing, C. N. Mathur, M. A. Haleem, R. Chandramouli, and K. Subbalakshmi, "Dynamic Spectrum Access with QoS and Interference Temperature Constraints," *Mobile Computing, IEEE Transactions on*, vol. 6, no. 4, pp. 423–433, 2007.

- [44] S. Gunawardena and W. Zhuang, "Service Response Time of Elastic Data Traffic in Cognitive Radio Networks," *Selected Areas in Communications, IEEE Journal on*, vol. 31, no. 3, pp. 559–570, 2013.
- [45] V. Mishra, L. C. Tong, and C. Syin, "QoS Based Spectrum Decision Framework for Cognitive Radio Networks," in *Networks (ICON), 2012 18th IEEE International Conference on*. IEEE, 2012, pp. 18–23.
- [46] —, "A QoS Provisioning MAC Protocol for Cognitive Radio Network," in *Signal Processing and Communication Systems (ICSPCS), 2013 7th International Conference on*. IEEE, 2013, pp. 1–8.
- [47] A. Homayounzadeh and M. Mahdavi, "Quality of Service Provisioning for Real-Time Traffic in Cognitive Radio Networks," *Communications Letters, IEEE*, vol. 19, no. 3, pp. 467–470, 2015.
- [48] N. Jiang and J. Hua, "QoS Performance Analysis for the Second User in the Overlay Cognitive Radio Networks," in *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2015 International Conference on*. IEEE, 2015, pp. 513–516.
- [49] S. Talat and L. Wang, "QoS-Guaranteed Channel Selection Scheme for Cognitive Radio Networks with Variable Channel Bandwidths," in *Proc. IEEE ICCAS'09*, July 2009, pp. 241–245.
- [50] T. Chakraborty, A. Mukhopadhyay, S. Bhunia, I. S. Misra, and S. K. Sanyal, "Analysis and Enhancement of QoS in Cognitive Radio Network for Efficient VoIP Performance," in *Information and Communication Technologies (WICT), 2011 World*

*Congress on.* IEEE, 2011, pp. 904–909.

- [51] T. Chakraborty, I. S. Misra, and S. K. Sanyal, “Selection of Optimal Transmission Time in Cognitive Radio Network for Efficient VoIP Performance,” in *Computers and Devices for Communication (CODEC), 2012 5th International Conference on.* IEEE, 2012, pp. 1–4.
- [52] E. Demirtaş and S. Oktuğ, “Providing QoS to Secondary Users Employing VoIP Applications in Cognitive Radio Networks,” in *ICSNC 2011*, 2011, pp. 83–87.
- [53] T. Chakraborty, I. Saha Misra, and S. K. Sanyal, “Proactive QoS Enhancement Technique for Efficient VoIP Performance over Wireless LAN and Cognitive Radio Network,” *Journal of Networks*, vol. 7, no. 12, pp. 1925–1942, 2012.
- [54] Y. Y. Mihov and B. P. Tsankov, “Cognitive System with VoIP Secondary Users over VoIP Primary Users,” in *Proceedings of the 3rd International Conference on Advanced Cognitive Technologies and Applications (COGNITIVE’11)*. Citeseer, 2011, pp. 30–35.
- [55] B. Kim, G.-m. Lee, and B.-h. Roh, “MAC Protocol for Quality-Aware Real-Time Voice Delivery in Cognitive Radio-Enabled WSNs,” *International Journal of Distributed Sensor Networks*, vol. 2015, 2015.
- [56] A. Homayounzadeh and M. Mahdavi, “Improving Voice-Service Support in Cognitive Radio Networks,” *ETRI Journal*, 2016.
- [57] P. Wang, D. Niyato, and H. Jiang, “Voice Service Support over Cognitive Radio Networks,” in *Communications, 2009. ICC’09. IEEE International Conference on.* IEEE, 2009, pp. 1–5.

- [58] H. Lee and D.-H. Cho, "VoIP Capacity Analysis in Cognitive Radio System," *Communications Letters, IEEE*, vol. 13, no. 6, pp. 393–395, 2009.
- [59] H. Lee and D. Cho, "Capacity Improvement and Analysis of VoIP Service in a Cognitive Radio System," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1646–1651, 2010.
- [60] H. Heffes and D. Lucantoni, "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance," *Selected Areas in Communications, IEEE Journal on*, vol. 4, no. 6, pp. 856–868, 1986.
- [61] Z. Wang, T. Jiang, L. Jiang, and X. He, "Voip Capacity Analysis in Cognitive Radio System with Single/Multiple Channels," in *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*. IEEE, 2010, pp. 1–4.
- [62] S. Gunawardena and W. Zhuang, "Voice Capacity of Cognitive Radio Networks," in *Proc. IEEE ICC'10*, May 2010, pp. 1–5.
- [63] —, "Voice Capacity of Cognitive Radio Networks for Both Centralized and Distributed Channel Access Control," in *IEEE Global Telecommunications Conference (GLOBECOM 2010)*. IEEE, 2010, pp. 1–5.
- [64] M. Schwartz, *Broadband Integrated Networks*. Prentice Hall, 1998.
- [65] A. I. Elwalid and D. Mitra, "Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 1, no. 3, pp. 329–343, 1993.

- [66] D. Wu and R. Negi, "Effective Capacity: A Wireless Link Model for Support of Quality of Service," *Wireless Communications, IEEE Transactions on*, vol. 2, no. 4, pp. 630–643, 2003.
- [67] K. Tan, K. Kim, Y. Xin, S. Rangarajan, and P. Mohapatra, "RECOG: A Sensing-Based Cognitive Radio System with Real-Time Application Support," *Selected Areas in Communications, IEEE Journal on*, vol. 31, no. 11, pp. 2504–2516, 2013.
- [68] K. Koufos, K. Ruttik, and R. Jantti, "Voice Service in Cognitive Networks over the TV Spectrum," *Communications, IET*, vol. 6, no. 8, pp. 991–1003, 2012.
- [69] S. Gunawardena and W. Zhuang, "Capacity Analysis and Call Admission Control in Distributed Cognitive Radio Networks," *Wireless Communications, IEEE Transactions on*, vol. 10, no. 9, pp. 3110–3120, 2011.
- [70] S. L. Castellanos-Lopez, F. A. Cruz-Pérez, M. E. Rivero-Angeles, and G. Hernandez-Valdez, "Impact of the Primary Resource Occupancy Information on the Performance of Cognitive Radio Networks with VoIP Traffic," in *Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM), 2012 7th International ICST Conference on*. IEEE, 2012, pp. 338–343.
- [71] —, "Performance Comparison of VoIP Cognitive Radio Networks Under On/Off and Poisson Primary Arrivals," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*. IEEE, 2013, pp. 3302–3307.
- [72] —, "VoIP Erlang Capacity in Coordinated Cognitive Radio Networks," in *Vehicular Technology Conference (VTC Fall), 2013 IEEE 78th*. IEEE, 2013, pp. 1–6.

- [73] —, “Joint Connection Level and Packet Level Analysis of Cognitive Radio Networks with VoIP Traffic,” *Selected Areas in Communications, IEEE Journal on*, vol. 32, no. 3, pp. 601–614, 2014.
- [74] S. L. Castellanos-Lopez, G. Hernandez-Valdez, M. E. Rivero-Angeles, and F. A. Cruz-Pérez, “Call Admission Control Strategy for Cognitive Radio Networks with VoIP-Traffic,” in *Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st*. IEEE, 2015, pp. 1–7.
- [75] L. Wang, A. Chen, and S. Wei, “A cognitive MAC Protocol for QoS Provisioning in Overlaying Ad hoc Networks,” in *Proc. IEEE ICCS’08*, Nov. 2008, pp. 1489–1493.
- [76] J. Chen, M. Zhao, J. Tian, and S. Li, “A QoS MAC Protocol for Cognitive PMP Networks with Rapid Changes of Spectrum Opportunities,” in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*. IEEE, 2008, pp. 1661–1665.
- [77] V. K. Tumuluru, P. Wang, D. Niyato, and W. Song, “Performance Analysis of Cognitive Radio Spectrum Access with Prioritized Traffic,” *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 4, pp. 1895–1906, 2012.
- [78] A. Alshamrani, X. Shen, and L.-L. Xie, “QoS Provisioning for Heterogeneous Services in Cooperative Cognitive Radio Networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 4, pp. 819–830, 2011.
- [79] S. C. Jha, U. Phuyal, M. M. Rashid, and V. K. Bhargava, “Design of OMC-MAC: An Opportunistic Multi-Channel MAC with QoS Provisioning for Distributed Cognitive Radio Networks,” *Wireless Communications, IEEE Transactions on*, vol. 10, no. 10, pp. 3414–3425, 2011.

- [80] T. Jiang, H. Wang, and A. V. Vasilakos, "QoE-Driven Channel Allocation Schemes for Multimedia Transmission of Priority-Based Secondary Users over Cognitive Radio Networks," *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 7, pp. 1215–1224, 2012.
- [81] R. Doost-Mohammady, M. Y. Naderi, and K. R. Chowdhury, "Spectrum Allocation and QoS Provisioning Framework for Cognitive Radio with Heterogeneous Service Classes," *Wireless Communications, IEEE Transactions on*, vol. 13, no. 7, pp. 3938–3950, 2014.
- [82] M. T. Masonta, F. Mekuria, M. Mzyece, and K. Djouani, "Adaptive Spectrum Decision Framework for Heterogeneous Dynamic Spectrum Access Networks," in *AFRICON, 2015*. IEEE, 2015, pp. 1–5.
- [83] T. Chakraborty, I. S. Misra, and T. Manna, "Design and Implementation of VoIP Based Two-Tier Cognitive Radio Network for Improved Spectrum Utilization," *IEEE Systems Journal*, vol. 10, no. 1, pp. 370 – 381, 2016.
- [84] H. Su and X. Zhang, "Secondary User Friendly TDMA Scheduling for Primary Users in Cognitive Radio Networks," in *Proc. IEEE GLOBECOM'09*, Nov. 2009, pp. 1–6.
- [85] H. Okii and I. A. Rai, "Providing Some Quality of Service for Secondary Users in Cognitive Radios Using Time Slotted Systems," in *e-Infrastructure and e-Services for Developing Countries*. Springer, 2011, pp. 24–34.

- [86] I. Suliman and J. Lehtomäki, "Queueing Analysis of Opportunistic Access in Cognitive Radios," in *Cognitive Radio and Advanced Spectrum Management, 2009. Cog-ART 2009. Second International Workshop on*. IEEE, 2009, pp. 153–157.
- [87] S. K. Jayaweera, G. Vazquez-Vilar, and C. Mosquera, "Dynamic Spectrum Leasing: A New Paradigm for Spectrum Sharing in Cognitive Radio Networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2328–2339, 2010.
- [88] J. M. Peha, "Approaches to Spectrum Sharing," *IEEE Communications Magazine*, vol. 43, no. 2, pp. 10–12, 2005.
- [89] Y. Han, S. H. Ting, and A. Pandharipande, "Cooperative Spectrum Sharing Protocol with Secondary User Selection," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 9, pp. 2914–2923, 2010.
- [90] Y. Yi, J. Zhang, Q. Zhang, and T. Jiang, "Spectrum Leasing to Multiple Cooperating Secondary Cellular Networks," in *Communications (ICC), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1–5.
- [91] Y.-W. Chan, F.-T. Chien, R. Y. Chang, M.-K. Chang, and Y.-C. Chung, "Spectrum Sharing in Multi-Channel Cooperative Cognitive Radio Networks: A Coalitional Game Approach," *Wireless networks*, vol. 19, no. 7, pp. 1553–1562, 2013.
- [92] C. Zhai, W. Zhang, and P. Ching, "Spectrum Sharing Based on Two-Path Successive Relaying," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2909–2912.

- [93] H. Xu and B. Li, "Efficient Resource Allocation with Flexible Channel Cooperation in OFDMA Cognitive Radio Networks," in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.
- [94] W. Su, J. D. Matyjas, and S. Batalama, "Active Cooperation between Primary Users and Cognitive Radio Users in Heterogeneous Ad-Hoc Networks," *Signal Processing, IEEE Transactions on*, vol. 60, no. 4, pp. 1796–1805, 2012.
- [95] I. Marić, R. D. Yates, and G. Kramer, "Capacity of Interference Channels with Partial Transmitter Cooperation," *Information Theory, IEEE Transactions on*, vol. 53, no. 10, pp. 3536–3548, 2007.
- [96] A. Jovičić and P. Viswanath, "Cognitive Radio: An Information-Theoretic Perspective," *Information Theory, IEEE Transactions on*, vol. 55, no. 9, pp. 3945–3958, 2009.
- [97] A. Attar, S. Ghorashi, M. Sooriyabandara, and A. Aghvami, "Challenges of Real-Time Secondary Usage of Spectrum," *Elsevier Journal on Computer Networks*, vol. 52, no. 4, pp. 816–830, 2008.
- [98] ITU-T, *G.107: The E-model, a Computational Model for Use in Transmission Planning*. ITU-T Recommendation, 2005.
- [99] R. Ahuja, T. Magnanti, and J. Orlin, "Network Flows: Theory, Algorithms, and Applications," *Journal of the Operational Research Society*, vol. 45, no. 11, pp. 1340–1340, 1994.
- [100] D. P. Bertsekas and R. G. Gallager, *Data Networks*, 2nd ed. Prentice Hall, 1992.

- [101] Voice Over IP - Per Call Bandwidth Consumption Manual. [Online]. Available: <http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth-consume.html>
- [102] R. L. Kruse, *Data Structures and Program Design*, 2nd ed. Prentice Hall, 1987.
- [103] AMPL: A Modeling Language for Mathematical Programming. [Online]. Available: <http://www.ampl.com/>
- [104] IBM ILOG CPLEX Optimizer. [Online]. Available: <http://www.ampl.com/DOWNLOADS/cplex80.html>.
- [105] Quality of Service Design Overview. [Online]. Available: <http://www.ciscopress.com/articles/article.asp?p=357102>
- [106] Z. Han, A. Kwasinski, and K. R. Liu, "A Near-Optimal Multiuser Joint Speech Source-Channel Resource-Allocation Scheme over Downlink CDMA Networks," *Communications, IEEE Transactions on*, vol. 54, no. 9, pp. 1682–1692, 2006.
- [107] J. Gordon, "Pareto Process as a Model of Self-similar Packet Traffic," in *Global Telecommunications Conference, 1995. GLOBECOM'95., IEEE*, vol. 3. IEEE, 1995, pp. 2232–2236.
- [108] V. Paxson and S. Floyd, "Why We Don't Know How to Simulate the Internet," in *Proceedings of the 29th conference on Winter simulation.* IEEE Computer Society, 1997, pp. 1037–1044.
- [109] N. I. Sarkar, "Impact of Traffic Arrival Distributions on an 802.11 Ad Hoc Network: Modeling and Performance Study," *Journal of Selected Areas in Telecommunications*, vol. 2, no. 5, pp. 9–16, 2012.

- [110] M. Grant, S. Boyd, and Y. Ye, “Cvx: Matlab software for disciplined convex programming,” 2008.
- [111] D. Niyato and E. Hossain, “Competitive Pricing for Spectrum Sharing in Cognitive Radio Networks: Dynamic Game, Inefficiency of Nash Equilibrium, and Collusion,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 1, pp. 192–202, 2008.
- [112] J. F. Hayes, *Modeling and Analysis of Computer Communications Networks*. Plenum, 1984.