



Déploiement à grande échelle de la voix sur IP dans des environnements hétérogènes

Abdelbasset Trad

► To cite this version:

Abdelbasset Trad. Déploiement à grande échelle de la voix sur IP dans des environnements hétérogènes. Networking and Internet Architecture [cs.NI]. Université de Nice Sophia Antipolis, 2006. English. NNT : . tel-00406513

HAL Id: tel-00406513

<https://theses.hal.science/tel-00406513>

Submitted on 22 Jul 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE NICE - SOPHIA ANTIPOLIS
UFR SCIENCES

École Doctorale STIC
Sciences et Technologies de l'Information et de la Communication

THÈSE DE DOCTORAT

Présentée par
Abdelbasset TRAD

en vue de l'obtention du titre de
DOCTEUR EN SCIENCES

de l'Université de Nice - Sophia Antipolis

Spécialité : INFORMATIQUE

Sujet de la thèse:

**Déploiement à Grande Échelle de la Voix sur IP dans des
Environnements Hétérogènes**

**Large Scale VoIP Deployment over Heterogeneous
Environments**

Laboratoire d'Accueil : INRIA Sophia-Antipolis
Projet de Recherche : Planète

Soutenue publiquement à l'INRIA le 21 Juin 2006 à 14h00 devant le jury composé de:

M.	Jean-Paul	Rigault	UNSA/INRIA	Président
M.	Hossam	Afifi	INT Evry/INRIA	Dir. de thèse
Mme	Houda	Labioud	ENST Paris	Rapporteur
M.	Pascal	Lorenz	Université de Haute Alsace	Rapporteur
M.	Tijani	Chahed	INT Evry	Examineur
M.	Ken	Chen	Université de Paris 13	Examineur

À mes grands-parents

À mes parents

À mon frère et à mes soeurs

Remerciements

Je voudrais remercier Mme Houda Labiod, Maître de conférences à l'ENST Paris et M. Pascal Lorenz, Professeur à l'Université de Haute Alsace, qui m'ont fait l'honneur d'avoir accepté d'être les rapporteurs de ma thèse. Je remercie également M. Jean-Paul Rigault, Professeur à l'Université de Nice - Sophia Antipolis, qui m'a fait l'honneur de présider mon jury de thèse. Je tiens aussi à remercier M. Tijani Chahed, Maître de conférences à l'INT Evry et M. Ken Chen, Professeur à l'Université de Paris 13, d'avoir bien voulu participer à ce jury. Leur déplacement pour ma soutenance était pour moi un grand honneur et un vrai plaisir.

Je tiens à exprimer mes remerciements les plus vifs à M. Hossam Afifi, Professeur à l'INT Evry, qui m'a encadré durant mon stage de fin d'études d'Ingénieur, mon stage de DEA et ma thèse de doctorat, j'ai appris de lui toute une philosophie de travail dans le domaine de la recherche scientifique. La liberté et la confiance qu'il m'a accordées ont beaucoup contribué au développement de mon autonomie dans le travail. Malgré qu'il n'était pas toujours près de moi, il m'a apporté lors de ses visites à l'INRIA Sophia Antipolis des conseils judicieux, des encouragements, un soutien moral, beaucoup de sympathie et de bonne humeur.

J'exprime mes remerciements et ma gratitude à M. Walid Dabbous, Directeur de Recherche et chef du projet Planète de l'INRIA Sophia Antipolis, pour la confiance et l'occasion qu'il m'a accordées afin de faire une grande partie de ma formation en recherche scientifique au sein du projet Planète. Les années que j'ai passées à Planète m'ont permis d'acquérir une expérience précieuse aussi bien sur le plan professionnel que sur le plan humain.

Merci individuellement à mes collègues de l'équipe Planète pour leur sympathie et pour l'ambiance amicale qui règne au sein du projet.

Un grand merci à M. Denis Caromel de m'avoir permis de concrétiser ma passion pour l'enseignement en donnant le cours "Multimedia sur l'Internet" pour les étudiants du Mastère Télécommunications de l'Université de Nice Sophia Antipolis, cette expérience d'enseignement était très enrichissante au niveau pédagogique et scientifique. Je dois pas oublier de remercier tous mes enseignants depuis l'école primaire, c'est eux qui ont éveillé en moi mon amour et ma passion pour la science et pour le savoir.

Sans doute je vais pas trouver les mots pour remercier les personnes qui me sont les plus chères: mes parents, pour leur sacrifices, leur patience, et tous ce qu'ils ont fait pour m'apporter le bonheur.

Enfin, je tiens à remercier tous mes amis et tous ceux et qui m'ont soutenu surtout pendant les moments délicats, en particulier les membres de ma petite famille qui m'ont accompagné par la pensée tout au long de cette thèse. J'ai également une pensée toute particulière pour mon grand-père décédé cette année. Par son exemple et ses valeurs, il a contribué à sa façon à l'aboutissement de ce travail que je lui dédie aujourd'hui.

Abdelbasset Trad

Sophia Antipolis, France.

DÉPLOIEMENT À GRANDE ÉCHELLE DE LA VOIX SUR IP DANS DES ENVIRONNEMENTS HÉTÉROGÈNES

RÉSUMÉ

Dans cette thèse, nous nous intéressons au déploiement à grande échelle de la Voix sur IP (VoIP) dans des environnements Internet hétérogènes. Après une description des mécanismes de codage et de transmission de la voix sur l'Internet, nous étudions dans une première partie de la thèse, les limites de performance dans le cas d'une transmission d'un grand nombre de flux de voix sur IP entre deux passerelles téléphoniques. Nous discutons le besoin d'utilisation de mécanismes de contrôle de congestion pour le trafic de voix sur IP qui est en croissance continue sur l'Internet. Nous proposons un nouveau schéma de contrôle de congestion de la voix sur IP. Ce schéma combine le multiplexage de flux RTP et le mécanisme de contrôle TCP-amical (TCP-friendly) afin d'améliorer l'efficacité et la performance de la transmission des flux de voix sur IP et de garantir l'équité avec les autres types de trafic coexistant sur l'Internet. La deuxième partie de la thèse est consacrée à l'étude de la transmission de la voix dans des environnements de réseaux locaux sans-fil IEEE 802.11e. Nous développons un modèle analytique permettant d'évaluer la capacité d'un réseau 802.11e en nombre de communications de voix sur IP en fonction des paramètres du niveau applicatif (codage audio utilisé) ainsi que des paramètres relatifs au canal de transmission sans-fil. Ce modèle peut être utilisé pour ajuster ces paramètres afin d'augmenter la capacité du réseau sans-fil tout en considérant les contraintes strictes des communications interactives de la voix sur IP. Dans la dernière partie de la thèse, nous étudions le cas de la transmission de la voix sur IP dans des environnements Internet hétérogènes (filaire/sans-fil) constitués en partie par des liens d'accès sans-fil. Nous proposons une architecture basée sur une passerelle de voix sur IP placée au bord du réseau sans-fil. Cette passerelle est utilisée pour adapter les flux de voix aux caractéristiques du réseau sans-fil. Le mécanisme d'adaptation proposé estime dynamiquement l'état de congestion du canal sans-fil et permet la différenciation entre les pertes de paquets causées par la congestion et celles dues aux erreurs de transmission sur le canal sans-fil. L'adaptation appropriée est alors appliquée. Le mécanisme d'adaptation proposé, ne nécessite pas de modifications du protocole de contrôle d'accès au canal sans-fil (MAC), ce qui facilite son déploiement sur l'infrastructure réseau existante.

MOTS CLÉS

Voix sur IP, déploiement à grande échelle, codecs audio, contrôle de congestion adaptatif, multiplexage de flux, contrôle TCP-amical, réseaux IEEE 802.11e, capacité réseau, E-Model.

LARGE SCALE VOIP DEPLOYMENT OVER HETEROGENEOUS ENVIRONMENTS

ABSTRACT

In this thesis, we focus on large scale Voice over IP (VoIP) deployment over heterogeneous Internet environments. We first present an overview of VoIP coding and transmission mechanisms. Then, we investigate performance limitations in the case of a large number of VoIP flows transported over wired Internet links between peer VoIP gateways. We address, in the first part of the thesis, the need to design congestion control for the growing class of VoIP traffic and we propose a novel VoIP congestion control scheme. This scheme combines RTP voice flow multiplexing and the TCP-friendly congestion control mechanism in order to improve VoIP transmission efficiency and performance while being fair with coexisting Internet traffic. The second part of this thesis deals with VoIP transmission over IEEE 802.11e wireless LAN environments (WLAN). We develop an analytical model to evaluate the capacity of an 802.11e network in terms of VoIP communications while conditioning on the used audio codec and MAC/PHY parameters. This model can be applied to tune IEEE 802.11e standard parameters in order to increase the WLAN capacity while considering stringent requirements of interactive VoIP communications. In the last part of this thesis we study the case of VoIP transmission over heterogeneous Internet environments where WLAN represents the last-hop Internet access link. We propose an architecture that is based on a VoIP gateway located at the edge of the wired and the wireless network, and used to adapt voice flows according to wireless network characteristics. The proposed adaptation mechanism dynamically estimates wireless channel congestion state and allows the differentiation between congestion loss and loss caused by wireless channel transmission errors. Then, the appropriate adaptation is applied. Specifically, in our proposed adaptive mechanism, the medium-access control (MAC) protocol at the wireless end stations does not need to be modified, making the mechanism more readily deployable over the existing network infrastructure.

KEYWORDS

Voice over IP, large scale deployment, audio codecs, adaptive congestion control, flow multiplexing, TCP-friendly rate control, IEEE 802.11e WLAN, network capacity, E-Model.

Contents

Résumé	i
Abstract	iii
Table of Contents	v
List of Figures	vii
List of Tables	xi
1 Introduction	1
1.1 Voice over IP Trends and Challenges	1
1.2 Motivations and Contributions of the Thesis	4
1.2.1 VoIP TCP-Friendly Congestion Control Mechanism (Voice-TFCC)	5
1.2.2 Capacity Evaluation of VoIP in IEEE 802.11e WLAN Environments	6
1.2.3 Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Internet Environments	7
1.3 Organization of the Manuscript	8
2 VoIP System Overview	9
2.1 Voice Coding	9
2.1.1 Voice Coding Techniques	10
2.1.2 Subjective Voice Quality	12
2.1.3 VoIP Standard Codecs	12
2.2 Transport of Real-time Voice over the Internet	15
2.2.1 RTP for Voice Media Transport	16
2.2.2 RTP Mixers and Translators	19
2.2.3 RTCP for Media Transmission Control and Management	19
2.3 Network Impairments and Proposed Improvements	24
2.3.1 End-to-End Delay	24
2.3.2 Jitter	26
2.3.3 Packet Loss	27
2.3.4 Echo Impairment	28

2.4	The E-Model for Real-Time Audio Quality Measurement	28
2.5	Conclusions	32
3	Towards a Congestion Control for VoIP Traffic	35
	Summary	35
3.1	Introduction	36
3.2	Problem Statement	37
	3.2.1 Studied Architecture	39
	3.2.2 Performance Limitations	39
3.3	Discussion of Related Work	40
	3.3.1 IP/UDP/RTP Header Compression	41
	3.3.2 RTP Multiplexing Schemes	42
	3.3.3 TCP-Friendly Rate Control for Unresponsive Flows	43
3.4	Proposed VoIP Congestion Control Scheme	48
	3.4.1 Voice-TFCC Scheme for RTP VoIP Flow Transmission	48
	3.4.2 Voice-TFCC Algorithm Description	50
3.5	Voice-TFCC Scheme Analysis	54
	3.5.1 Delay and Loss Rate of Voice-TFCC Packets	54
	3.5.2 Discussion of the Variable Size of Voice-TFCC Packets	55
	3.5.3 Comparison of Voice-TFCC Scheme with Related Schemes	55
3.6	Evaluation of the Voice-TFCC Scheme	56
	3.6.1 Saving Bandwidth by Voice-TFCC Scheme	56
	3.6.2 Experimental Results and Discussion	58
3.7	Conclusions	68
4	Capacity Evaluation of VoIP in IEEE 802.11e WLAN Environments	71
	Summary	71
4.1	Introduction	72
4.2	IEEE 802.11 MAC Standard	72
4.3	Analysis of IEEE 802.11e QoS Enhancements	74
	4.3.1 HCF Controlled Channel Access (HCCA)	75
	4.3.2 Enhanced Distributed Coordinated Access (EDCA)	76
4.4	Related Work	78
4.5	Studied Network System	80
4.6	Protocol Layers and Headers Overhead	80
4.7	VoIP Capacity in IEEE 802.11e Networks	81
	4.7.1 Proposed VoIP Capacity Model	82
	4.7.2 Modeling Results for VoIP Capacity under HCCA/EDCA	83
	4.7.3 Effect of RTS/CTS protection mode on VoIP Capacity	90
	4.7.4 VoIP Capacity with VBR Sources	91
4.8	Conclusions	91

5 Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks	93
Summary	93
5.1 Introduction	94
5.2 Related Work	95
5.2.1 Rate Control	95
5.2.2 Loss Recovery	96
5.2.3 Wireless and Congestion Loss Differentiation	97
5.3 Problem Statement	98
5.4 Proposed Architecture and Rate Control Mechanism	99
5.4.1 VoIP Gateway Inter-working Wired and Wireless Network	99
5.4.2 TCP Vegas Congestion Control Mechanism	100
5.4.3 Vegas-Like Audio Rate and Loss Control Mechanism	101
5.5 Simulation Experiments and Discussion	103
5.6 Conclusions	107
6 Conclusions and Perspectives	109
Bibliography	112
Glossary	120
Annex A	123
Présentation des Travaux de Thèse en Français	127

List of Figures

1.1	Convergence towards packet data networks	2
2.1	Subjective codec quality: effect of coding technique and codec bit rate	13
2.2	RTP packet encapsulation	15
2.3	RTP header format	17
2.4	RTP mixer	19
2.5	RTCP sender and receiver reports sent conjointly with RTP packets	20
2.6	Packet format of the RTCP Receiver Report	21
2.7	Packet format of the RTCP Sender Report	21
2.8	RTT estimation	23
2.9	R-factor and MOS mapping	29
2.10	MOS vs. intrinsic codec Impairment factor I_e and loss rate	31
2.11	MOS of the G.711 codec as a function of network delay and mean loss rate	31
2.12	MOS of the G.723.1 and G.729 codec as a function of network delay and mean loss rate	32
3.1	Voice packet encapsulation	38
3.2	Protocol header overhead according to codec payload size	38
3.3	Heterogeneous mix of data and real-time services transmitted on the same IP infrastructure	39
3.4	TFRC throughput as function of drop rate and RTT (payload size = 1460 bytes)	44
3.5	Rate reduction factor decreasing for small payloads	46
3.6	Drop rates and sending rates of TCP and VoIP TFRC flows in a drop-tail packet-based queue environment	47
3.7	Drop rates and sending rates of TCP and VoIP TFRC flows in a drop-tail byte-based queue environment	47
3.8	Packets from different RTP flows multiplexed into one UDP packet	48
3.9	Adjusting the multiplexing buffer size in response to traffic fluctuations using the TCP-friendly congestion control mechanism	49
3.10	Voice-TFCC gateway architecture	49
3.11	Flow chart of the Voice-TFCC scheme	54
3.12	Bandwidth saving as the number of multiplexed packets increases	57

3.13 Bandwidth saving as the size of multiplexed packets increases	58
3.14 MOS as function of network delay and codec Impairment factor	59
3.15 Traceroute RTT measurement for network hops between source and destination hosts	60
3.16 Packet delay of (a) VoIP flows vs. packet delay of (b) Voice-TFCC flow	61
3.17 Jitter of (a) VoIP flows vs. jitter of (b) Voice-TFCC flow	61
3.18 MOS of (a) VoIP flows vs. MOS of (b) Voice-TFCC flow	62
3.19 Rate difference between actual VoIP transmission rate and TCP-friendly rate estimation	62
3.20 Packet delay of (a) VoIP flows vs. packet delay of (b) Voice-TFCC flow	63
3.21 Jitter of (a) VoIP flows vs. jitter of (b) Voice-TFCC flow	63
3.22 MOS of (a) VoIP flows vs. MOS of (b) Voice-TFCC flow	64
3.23 Rate difference between actual Voice-TFCC transmission rate and TCP-friendly rate estimation	64
3.24 Packet delay of Voice-TFCC flow	66
3.25 Jitter of Voice-TFCC flow	66
3.26 MOS of Voice-TFCC flow	67
3.27 Number of multiplexed VoIP flows	67
3.28 Rate difference between actual Voice-TFCC transmission rate and TCP-friendly rate estimation	67
4.1 IEEE 802.11 DCF medium access scheme	74
4.2 Alternation of CFP, CP and triggered CAP during an IEEE 802.11e superframe	75
4.3 Timing diagram to support QoS according to AC in EDCA	77
4.4 Wireless network scenario	80
4.5 DCF overhead and data frame	81
4.6 MOS of standard codecs as function of delay	84
4.7 Improved maximum number of supported voice calls under EDCA/HCCA	85
4.8 VoWLAN (a) packet delay and (b) MOS as function of the number of supported VoIP calls	86
4.9 Decreased VoIP capacity under EDCA/HCCA when CP duration is increased	87
4.10 Maximum number of supported voice calls under (a) HCCA/(b) EDCA	87
4.11 Comparison of VoIP capacity under 802.11b/g for (a) G.711 (b) G.729 and (c) G.723.1	88
4.12 Maximum number of supported voice calls under HCCA/EDCA as function of increas- ing 802.11b/802.11g physical transmission rate	89
4.13 Comparaison of the maximum number of supported voice calls under EDCA when RTS/CTS option is (a) disabled/ (b) enabled	90
5.1 Voice traffic transmitted from wired network through a last hop congested wireless link	99
5.2 VoIP gateway at the edge of wired and wireless network	100
5.3 Flow-chart for the Vegas-like audio rate control	101
5.4 Mean VoIP delay vs. the number of QSTAs	104
5.5 Maximum VoIP delay vs. the number of QSTAs	105

5.6	Mean bandwidth of VoIP flows vs. the number of QSTAs	105
5.7	Improved VoIP flow fairness with the adaptive Vegas-like codec rate control	106
6.1	ON-OFF modeling of VoIP traffic	125

List of Tables

2.1	MOS subjective quality scores and correspondent impairment	12
2.2	Features of main standard VoIP codecs	15
2.3	Tolerable delay limits in the absence of packet loss	25
2.4	Tolerable packet loss rates for delay values less than 150 ms	27
2.5	Intrinsic equipment Impairment factor I_e	30
2.6	Packet-loss robustness factor B_{pl}	30
3.1	Low bit rates generated by three standard IP telephony speech codecs	37
3.2	PlanetLab hosts used for experimental study	58
3.3	Features of main standard VoIP codecs	59
3.4	Results summary: with and without Voice-TFCC mechanism	65
3.5	Results summary: Voice-TFCC mechanism without codec rate adaptation	68
4.1	Parameter values of IEEE 802.11a/b/g physical layer standards	73
4.2	EDCA default parameter set	78
4.3	System modeling parameters	84
4.4	Codec features	84
4.5	Maximum number of VoIP calls for standard voice for different 802.11b channel transmission rates	89
4.6	Maximum number of VoIP calls for standard voice codecs as function of 802.11g data transmission rates	89
4.7	Effect of RTS/CTS on VoIP capacity under EDCA for $T_{SF} = 120$ ms	90
5.1	Codec bit rate and packet size	103
5.2	Simulation system parameters	104

Chapter 1

Introduction

In this thesis, we focus on large scale Voice over IP (VoIP) deployment over heterogeneous Internet environments. In the first part of the thesis we design a novel congestion control mechanism for the growing class of VoIP traffic. In a second part, we develop an analytical model to evaluate the capacity of an 802.11e network in terms of VoIP communications. In a third part, we propose an adaptive architecture to improve VoIP transmission performance over heterogeneous wired/wireless environments.

1.1 Voice over IP Trends and Challenges

The first wide area networks (WANs) were circuit-switched telephone networks. In these networks, the number of simultaneous users is limited, but when a connection is established, service is guaranteed for the total duration of the connection for each user. Although the service guarantee causes resources to be reserved once they have been claimed by a user, full utilization of network resources is difficult to attain in telephone networks. For this reason, today's inter-networks are mostly best-effort networks. In these store-and-forward networks, the number of simultaneous users is unlimited, but service is not guaranteed. Data packets are sent from the source to intermediate routers. A router stores the packets until they can be transferred to the next router. This continues in succession until the packets reach their destination. The flexibility of IP-based packet switched networks and the explosive growth of the Internet technology and applications based on the Internet Protocol (IP) have pushed the convergence of data (packet switched), and voice (traditionally circuit switched) into a single IP-based core architecture. The Internet has become an ubiquitous means of communication, and the total amount of packet-based network traffic has quickly surpassed traditional circuit-switched voice network traffic (Figure 1.1)¹.

In the wake of these technology advancements, Voice over IP also known as IP telephony has become a key driver in the evolution of voice communications. It has become clear that voice traffic and services will be one of the next major applications to take full advantage of IP networks

¹DS0 (Digital Signal 0): a basic standard digital transmission rate of 64 kbps, corresponding to the capacity of one voice-frequency-equivalent channel.

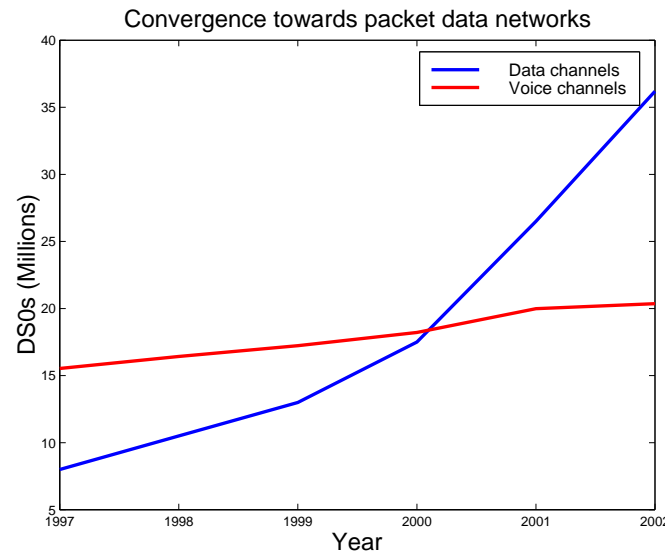


Figure 1.1: Convergence towards packet data networks

[58]. Attention is now being focused on IP telephony systems as networks that will replace the telephone network and become the base for the next generation of multimedia communications where a single converged network for the transport of voice and data will be used. VoIP technology is useful not only for phones but also as a broad application platform enabling voice interactions on devices such as PCs, mobile handhelds, and many vertical-specific application devices where voice communication is an important feature. Indeed, VoIP supplies many unique capabilities to the carriers and users who depend on IP or other packet-based networks.

While initial interest has come from large enterprises wishing to reduce their communication costs and from users looking for cheap long-distance and international telephone calls, consumer market has taken the lead now. In the last two years, VoIP has moved from hype to reality and VoIP services are being increasingly offered to end customers in the form of a low cost and easy to use service (e.g., skype tool [4]). The main benefits of VoIP technology include the following:

- Cost savings

By moving voice traffic to IP networks, companies can reduce or eliminate the toll charges associated with transporting calls over the Public Switched Telephone Network (PSTN). Service providers and end users can also conserve bandwidth by investing in additional capacity only when it is needed. This is made possible by the distributed nature of VoIP and by reduced operations costs as voice and data traffic can be multiplexed onto one network.

- Efficiency

First, the connectionless transmission mode where bandwidth is consumed only when voice packets are delivered, achieves more efficiency than the circuit-switched voice transmission. In addition, by exploiting advanced voice-compression techniques and bandwidth sharing in packet-switched networks, VoIP can dramatically improve bandwidth efficiency.

- Integration of voice and data networks

By making voice «just an IP application», integrated networks for voice and data can be built. These integrated networks not only intend to provide the voice transmission quality and reliability of PSTN networks, they also enable to quickly and flexibly create new services that combine voice communication with other media and data applications (e.g., video).

Given interactivity requirements, VoIP represents the most sensitive case of real-time multimedia traffic. In particular the delay is most critical in VoIP applications. VoIP transmission has unacceptable performance if long delays are incurred. Recommendation G.114 [3] of the International Telecommunications Union (ITU-T)² indicates that the end-to-end delay has a great impact on the perceived quality of interactive VoIP conversations with a threshold effect around 150 ms. Therefore, current best-effort IP network with highly variable delays will not be always sufficient for high quality voice delivery. In fact, best-effort IP networks were originally designed to support elastic data applications (that have no stringent delay requirements). These networks do not provide any guarantees regarding the delivery of packets, including whether or not they will be delivered at all. Most routers in the current Internet employ FIFO queue management coupled with a drop-tail buffer management policy. The packets are sent out in the order in which they are received, and any packet that arrives at a full buffer is dropped. In addition, routers make no attempt to notify the sender either about incipient congestion or about the dropped packets. Two main approaches have been proposed in the literature in order to guarantee a certain QoS parameters (i.e., bandwidth, delay and packet loss rate) to data flows containing time sensitive information (e.g., coded voice). The first approach consists in modifying the network architecture in order to provide better quality of service. Considering this approach, two major architectural network QoS models have been developed within the IETF: the Integrated Services (IntServ) [5] and the Differentiated Service (DiffServ) [7]. IntServ uses the Resource Reservation Protocol (RSVP) [6] to ensure end-to-end per-flow bandwidth reservations at each router along the path from the source to the destination of each flow. However, this architecture requires maintenance of individual flow states in the routers and its signaling complexity grows with the number of users. Therefore, such architecture may not scale well [2]. The scale limitations have led the IETF to consider the DiffServ architecture that provides a more lightweight approach to network QoS. This architecture ensures different classes of service with some performance guarantees to the heterogeneous aggregation of Internet flows. Nevertheless, the queuing strategy of DiffServ tends to break down in the case of a large number of flows. Due to their complexity and scalability problems, none of these network QoS architectures have yet seen widespread deployment [13]. The network QoS approach is out of scope of this thesis.

The second approach tries to improve quality of service without making changes to the network infrastructure based on end-to-end application-level QoS mechanisms. These end-to-end mechanisms do not rely on explicit support from the network beyond normal packet transport [38]. The end-to-end approach includes modifying the application implementations to make them more adap-

²<http://www.itu.org>

tive to variations in packet delays and losses, e.g., end systems measure the service being delivered by the network (i.e., using RTCP [9] reports), and use measurement information to adapt their behavior appropriately. One of the primary mechanisms used for end-to-end compensation for packet loss is Forward Error Correction (FEC) [11][12], but this technique introduces an additional delay due to the waiting time for the redundant information. Adaptive playout buffer algorithms [10] can be used for jitter compensation but this again is accomplished at the expense of end-to-end delay [12]. Alleviating losses and jitter is therefore a feasible task. However, the heterogeneity growth of current Internet leads us to conclude that it is unlikely for any end-to-end solution to be ubiquitously deployed. For example, VoIP packets should flow through high bandwidth core network links, even in peak traffic conditions, without any queuing delays and elaborate protocols to improve VoIP performance are not necessary in that case. Nevertheless, in some cases, access links may be expensive and broadband access difficult to obtain such as in WLAN environments. Moreover, over provisioning is not as straightforward as it may appear since adding capacity can sometimes lead to a delay increase for each and every user in congested networks as stated in Braess paradox [14]. Hence, QoS mechanisms are required to improve performance on specific parts of network paths.

Although we do not consider network-level QoS approach in this thesis, the mechanisms that we propose are compatible with network QoS mechanisms and can be used in conjunction to further enhance VoIP performance.

1.2 Motivations and Contributions of the Thesis

Research work on carrying voice over IP-based networks started in 1977 [1]. The contributions that followed, focused on enhancing the low voice transmission quality related to the intrinsic properties of IP packet networks originally designed to support elastic data traffic (asynchronous delivery, high packet loss rates, highly variable network delays, substantial packet jitter). The goal was to offer at least the circuit-switched network quality for VoIP communications. Proposed solutions particularly focused on recovery from packet jitter through the use of receiver jitter buffers, and loss recovery techniques to compensate packet loss [38]. These solutions studied the case of single VoIP communications and have not considered scale issues.

Currently, VoIP is one of the fastest growing Internet applications. VoIP traffic is then expected to have more rapid growth due to the large VoIP deployment. Simultaneously, the Internet heterogeneity is increasing due mainly to the fast deployment of wireless local area networks (WLANs). The scale in number of VoIP communications and the heterogeneity growth of current Internet infrastructure have created new performance limitations for VoIP deployment. The main motivation behind this work is to propose solutions and improvements to tackle these new limitations.

In this thesis, we mainly focus on the problem of large scale VoIP deployment over heterogeneous IP networks. We address the tradeoffs between efficiency and end-to-end overall performance of large number of VoIP communications. Our goal is to show how high quality of large scale VoIP transmission can be obtained by using adaptive congestion control mechanisms that dynamically

adjust their behavior according to the network congestion state in order to:

- Maximize the overall VoIP transmission quality.
- Use network resources efficiently.
- Compete fairly with other Internet traffic (i.e., TCP traffic).

Because of heterogeneity growth of the Internet, end-to-end solutions are getting less effective. Therefore, we base our architectural proposal on intermediary nodes (i.e., VoIP gateways) implementing adaptive control mechanisms. These intermediaries can take an active role in the gathering of report feedbacks from VoIP end-points and the diagnostics of network congestion state and then in the execution of the appropriate control mechanism on VoIP flows.

Thesis contributions are summarized here and detailed in the following chapters of the dissertation.

1.2.1 VoIP TCP-Friendly Congestion Control Mechanism (Voice-TFCC)

Given the continuous VoIP deployment, the Internet is expected to carry a significant proportion of the world's telephony traffic. Typically, voice traffic is deployed as best-effort traffic over Internet links. However, this best-effort voice traffic lacks effective and scalable end-to-end congestion control. These reasons led us to focus, in this contribution, on the case of a large number of VoIP sources at an access network, sharing a common path over IP backbone networks and destined to different users in remote networks. For example, long distance VoIP calls, including calls that are serviced by a combination of a switched telephone network in the local area and the Internet for the long haul.

We address, in this part of the thesis, the need to design congestion control for the growing class of VoIP traffic. This need is motivated by the inefficient use of network bandwidth caused by protocol header overhead of voice packets and the fairness problem caused by the transmission of large number of small VoIP packets sharing network links with TCP traffic.

We propose a novel VoIP congestion control scheme called Voice-TFCC (Voice TCP-Friendly Congestion Control), which tries to keep the transmission protocol overhead to a minimum. Voice-TFCC combines RTP voice flow multiplexing, codec rate adaptation and the TCP-friendly congestion control mechanism. The approach developed in this contribution is to adapt the transport protocol and the way in which it interacts with the network in order to support voice flows competing with TCP traffic. Voice-TFCC mechanism is applied on VoIP flows transmitted between two intermediate gateways. For a given estimation of the network congestion level, it adjusts the packet and codec rate in a TCP-friendly manner. Flow multiplexing is used to reduce the traffic load (i.e., packet rate) on Internet routers and to reduce the overall header bandwidth used by small VoIP packets. That is, based on network feedback information reported from the destination gateway, the sender gateway would reduce the number of multiplexed RTP packets during normal load situations and increase it otherwise. When packet multiplexing limits given by the MTU (Maximum Transfer Unit) size are

reached, Voice-TFCC will adapt codec rate (i.e., voice packet size) in addition to packet rate. To the best of our knowledge our proposal is the first scheme that incorporates packet and codec rate adaptation for RTP flows while maintaining TCP-friendliness.

We used analytical results to show the bandwidth efficiency obtained through our proposal. We also implemented a prototype version of Voice-TFCC scheme and evaluated its performance using wide area Internet experiments over PlanetLab [77] network. Experimental results show that Voice-TFCC scheme achieves efficient VoIP flow transmission and improves voice transmission quality while being fair to TCP, especially when used in the case of large number of VoIP flows. We believe that Voice-TFCC scheme combined with accurate estimation of network congestion state provides a significant step towards an efficient and scalable congestion control mechanism for VoIP traffic.

1.2.2 Capacity Evaluation of VoIP in IEEE 802.11e WLAN Environments

Due to the recent advances in wireless networking technology, wireless local area networks (WLANs) are fast becoming the “last-mile” of choice for the overall Internet infrastructure. Most business offices, universities and airports now use WLANs as Internet access links. WLANs hold the promise of providing unprecedented mobility, flexibility and scalability than their wired counterparts. At the same time VoIP is spreading rapidly especially in public spaces given the advantages of VoIP technology: mainly the significant improvement of bandwidth efficiency, communication costs reduction and the service flexibility. Because of the convergence of these two trends, VoIP over WLAN (VoWLAN) is intended to become an important Internet application. Therefore, WLANs will need to support a large number of concurrent VoIP communications.

For these motivations, we study in this part of the thesis VoIP capacity in 802.11e WLAN environments. This capacity is defined as the maximum number of simultaneous VoIP communications supported over the WLAN. We first discuss the negative effects of protocol layers and headers overhead on the WLAN capacity. We propose an analytical model for VoIP capacity evaluation under HCF (EDCA/HCCA) mode of the IEEE 802.11e standard [86]. Using modeling results, we address the influence of application level parameters (i.e., voice codec) as well as MAC/PHY parameters on the WLAN capacity. We illustrate performance results relative to typical codec data rates of G.711, G.729 and G.723.1 standard codecs. We also compare the VoIP capacity under EDCA and HCCA modes. We find that low bit rate codecs (i.e., G.729 and G.723.1) significantly increase VoIP capacity in comparison to G.711 codec constrained by throughput. We discuss the trade-off between the WLAN capacity (mainly dictated by MAC/PHY parameters) and the VoIP communication quality. We show that optimal VoWLAN quality can be obtained by the use of codec adaptation mechanism that determines the voice codec to be used according to the number of supported voice calls.

1.2.3 Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Internet Environments

Current Internet infrastructure is interconnecting heterogeneous networks (fixed and mobile, high and low bandwidth) and supporting a wide area of applications. The Internet heterogeneity is mainly increasing due to the fast deployment of wireless local area networks. This fast deployment of WLANs has generated a lot of interest in making VoIP available over such networks. In order for this to become reality, several issues related to VoIP transmission over heterogeneous wired/wireless networks should be solved.

In WLAN, VoIP traffic performance can be compromised severely because of the extreme variability of wireless channels, and also because of data traffic from other coexisting non-real time applications (e.g., ftp, Web). Implementing quality of service (QoS) mechanisms was the basic solution to support multimedia traffic, in particular VoIP traffic, over WLAN. The goal was to improve current standards with additions that take into account the different requirements of regular data traffic and time sensitive multimedia traffic (e.g., voice, video). The IEEE 802.11 working groups have extended 802.11 MAC layer to provide QoS support (i.e., IEEE 802.11e standard). However, even if QoS mechanisms are implemented, it is clear that WLANs will continue to offer much more challenging conditions than typical wired LANs. QoS should be combined with other mechanisms that take into consideration the high variability of wireless network conditions. Moreover, efficient adaptive mechanisms are required to alleviate the effect of network heterogeneity on VoIP transmission performance. Actually, the challenges in deploying VoIP over WLAN stem mainly from issues related to access point congestion and to those that affect the link quality. Typically, WLANs experience significantly higher delay, with more jitter and packet loss than wired networks especially when several users are connected to the same access point. Congestion easily occurs and the efficiency of the system quickly deteriorates when the number of users increases. In addition, due to intrinsic properties, wireless link layer provides only partial reliability. With the presence of wireless links, a large percentage of packets are lost due to transmission errors on these links. At the same time, in case of high load conditions, packets may be lost because of congestion. If wireless losses are misclassified by end-systems as congestion losses, the available bandwidth will not be used efficiently and this will bring down VoIP performance. On the other hand, if congestion losses are misclassified as wireless, generated flows may react not appropriately and cause more congestion in the network. The aim of loss differentiation schemes is then to differentiate between wireless and congestion losses in order to behave correctly and efficiently.

In this part of the thesis published in [35], we propose an adaptive architecture for the transport of VoIP traffic over heterogeneous wired/wireless Internet environments. This architecture uses a VoIP gateway associated with an 802.11e QoS enhanced access point (QAP) to transcode voice flows before their transmissions over the last-hop wireless link. The instantaneous bit rate is determined by a control mechanism based on the estimation of wireless channel congestion state. Our mechanism dynamically adapts audio codec bit rate using a delay-based congestion avoidance technique so as to preserve acceptable levels of quality especially during high network load conditions. The proposed adaptation mechanism also allows the differentiation between congestion

and wireless losses in order to respond appropriately (reduce coding rate/add FEC redundant information). The performance of the proposed mechanism was evaluated through extensive NS-2 [37] simulations and compared with TFRC mechanism. Simulation results show that the proposed mechanism improves voice transmission performance, especially when the number of stations is fairly large. The system capacity is also increased, since the sending rate is reduced in case of high-load network conditions. Obtained results also show that our adaptive rate control mechanism is fairer than TFRC mechanism especially in the case of large number of VoIP flows. Proposed adaptive mechanism allows the harmonious coexistence of VoIP and TCP without introducing modifications of the 802.11 MAC layer, making the mechanism more readily deployable over the existing network infrastructure.

1.3 Organization of the Manuscript

The remainder of this manuscript is organized as follows. First, we overview VoIP system in Chapter 2. Our contributions related to congestion control mechanisms for VoIP traffic appear in Chapter 3. Our contributions related to VoIP transmission over heterogeneous wired/wireless Internet environments appear in Chapter 4 and Chapter 5.

In Chapter 2, we overview the principal components of VoIP system related to the coding and transmission of voice media over IP networks and we detail main standard VoIP protocols. We also describe the ITU-T E-Model used for objective VoIP quality estimation.

In Chapter 3, we argue the need to design congestion control for the growing class of VoIP traffic and we introduce a new VoIP congestion control scheme which combines RTP voice flow multiplexing and the TCP-friendly congestion control mechanism in order to adapt packet and codec rate of VoIP flows while being fair with coexisting Internet traffic sharing the same network links (i.e., TCP traffic).

In Chapter 4, we develop an analytical model to evaluate the capacity of IEEE 802.11e wireless network in terms of the number of simultaneous VoIP communications depending on the used codec. Then, we discuss the effect of 802.11e MAC/PHY parameters on this capacity.

In Chapter 5, we present an adaptive architecture for the transport of VoIP traffic over heterogeneous wired/wireless Internet environments. This architecture uses a VoIP gateway associated with an IEEE 802.11e QoS enhanced access point (QAP) to dynamically adapt VoIP codec bit rate according to the wireless channel congestion state in order to improve VoWLAN performance. In order to reduce packet delays and losses, the adaptation is based on a congestion avoidance technique and a loss prediction mechanism that allows the differentiation between wireless and congestion losses.

The final chapter concludes the dissertation by reviewing the main contributions and by giving an outlook on future research directions.

Chapter 2

VoIP System Overview

Introduction

Internet telephony differs in a number of aspects from the PSTN, both in terms of architecture and protocols. Fundamentally, IP telephony networks relies on the end-to-end paradigm for delivery of services. In these networks, signaling and media transport protocols are between the end-systems involved in the VoIP call. Network routers treat media and signaling packets while ignoring any semantics implied by them. Indeed, one of the largest advantages of Internet telephony compared to the PSTN is the transparency of the network to the media carried.

In this chapter, we review the principal components of VoIP system related to the coding and transmission of voice media over best-effort IP networks. First, we study the principles involved in voice coding and we give details of commonly used voice codecs in Section 2.1. Secondly, we detail standard protocols used for the transport of real-time voice traffic over the Internet in Section 2.2. In Section 2.3, we investigate VoIP quality impairments caused by the IP network and we discuss the major improvement mechanisms proposed in literature. Finally, in Section 2.4 we present the E-Model used for VoIP objective quality measurement.

2.1 Voice Coding

Voice coding is necessary to convert the analog speech signal to digital packet voice. This often includes compression and decompression (codec) of digital signals to low bit rates to meet the bandwidth requirements of efficient voice transport across both the network and the subscriber's "last mile" link. Although there are many possible voice sounds which can be produced, voice can be considered to be quasi-stationary over short periods of time (of the order of 20 ms) [15]. Audio coders attempt to exploit the high predictability of voice signals in order to reduce the data rate necessary for good quality voice transmission. An ideal voice codec¹ will represent the voice with as few bits as possible, while producing reconstructed voice which sounds almost identical, to the original voice.

¹the terms codec and coder are used interchangeably

2.1.1 Voice Coding Techniques

We can distinguish three main categories of codec techniques: waveform codecs, source codecs, and hybrid codecs. Typically waveform codecs are used at high bit rates, and give very good voice quality. Source codecs operate at very low bit rates, but tend to produce synthetic voice. Hybrid codecs use techniques from both source and waveform coding, and give good quality voice at intermediate bit rates.

2.1.1.1 Waveform Codecs

Waveform codecs operate without using any knowledge of how the signal to be coded was generated. Generally they are low complexity codecs and produce high quality voice at rates above 16 kbits/s. When the data rate is lowered below this level the reconstructed voice quality that can be obtained degrades rapidly. Pulse Code Modulation (PCM) codecs are the simplest form of waveform codecs. For coding voice it was found that by using non-linear quantization, 8 bits per sample was sufficient in order to have voice quality which is almost indistinguishable from the original. Typical narrow-band voice sampling at 8 kHz gives a PCM bit rate of 64 kbits/s. A commonly used technique in waveform voice coding is to predict the value of the next sample ($n+1$) from the previous samples (n , $n-1$, $n-2$, etc.) because of correlations present in voice samples. If the predictions are effective then the error signal between the predicted samples and the actual samples will have lower variance than the original speech samples. Therefore, this error signal could be quantized with fewer bits than the original signal [15]. This is the basis of Differential Pulse Code Modulation (DPCM) schemes. At the decoder the quantized difference signal is added to the predicted signal to give the reconstructed voice signal. Adaptive Differential PCM (ADPCM) improved DPCM technique by using adaptive predictor and quantizer, so that they change to match the characteristics of the voice being coded.

2.1.1.2 Source Codecs

Source codecs or vocoders operate using a model of how the source was generated, and attempt to extract, from the signal being coded, the parameters of the model. Only simplified parametric information is transmitted to the decoder. Voice is represented as a time-varying filter and is excited with either a white noise source, for unvoiced segments, or a train of pulses separated by the pitch period for voiced segments². Source codecs require low bandwidth around 2.4 kbits/s, and produce voice which although intelligible is far from natural sounding. Increasing the bit rate much beyond 2.4 kbits/s is not worthwhile because of the inbuilt limitation in the coder's performance due to the simplified voice production. The main use of these codecs has been in military applications where natural sounding voice is not as important as a very low bit rate to allow heavy protection and encryption.

²Speech sounds are classified into voiced/unvoiced segments depending on their mode of excitation.

2.1.1.3 Hybrid Codecs

Hybrid codecs attempt to fill the gap between waveform and source codecs. As described above, waveform codecs are capable of providing good quality voice at bit rates down to about 16 kbits/s. Source codecs on the other hand can provide intelligible voice at 2.4 kbits/s, but cannot provide natural sounding voice. The most successful and commonly used hybrid codecs are the Analysis-By-Synthesis (ABS) codecs. Such codecs use the same linear prediction filter model of the vocal tract as found in source codecs. However instead of applying a simple two-state, voiced/unvoiced, model to find the necessary input to this filter, the excitation signal is chosen by attempting to match the reconstructed voice waveform as closely as possible to the original voice waveform. In Linear Predictive Coding (LPC), the predicted signal value at time n is given by:

$$S_p(n) = \sum_{k=1}^{k=p} a_k S(n-k) \quad (2.1)$$

The coefficients a_k must be chosen to approach the $S(n)$ value. If the prediction is accurate then the difference between the real and predicted voice samples $e(n) = S(n) - S_p(n)$ will have a lower variance than the real voice samples. Therefore, this error signal could be quantized with fewer bits than the original voice signal. So the issue of transmitting the speech information is translated to the transmission of the set of coefficients a_k . Coefficients a_k can be fixed or adaptive, i.e. computed for each new sample. The calculation of the set a_k requires a very high computational task of an inversion matrix to solve a set of linear equations [16]. Code-Excited Linear Predictive (CELP) and the Regular-Pulse Excited (RPE) codecs are extensions of the LPC coding. At bit rates lower than 16 kbits/s, CELP codecs and their derivatives, tend to be used. However, because of the forward adaptive determination of filter coefficients used in most of these codecs, they tend to have high coding delays that will contribute in increasing overall end-to-end delay of VoIP transmission and thus can cause voice quality degradation.

2.1.1.4 Silence Suppression

Typical voice conversations contain up to 65% of silence [16]. Reduction in the codec bit rate can be achieved by means of silence suppression³, in which case no signal is encoded during silence periods. This feature saves more than the half of required bandwidth. Silence suppression includes three main components: voice activity detection (VAD), discontinuous transmission (DTX), and comfort noise generation (CNG). VAD is responsible for determining when the talker is silent. DTX stops transmitting frames when the VAD has detected a silent period. CNG is used to recreate low-level background noise to the receiver. CNG is recommended for user confidence in the call connection: if the call appears too quiet, users may anticipate that the call has been disconnected. When silence suppression is employed, the codec then operate in two states: a silent state at zero

³The term silence compression is also used in the literature

bit-rate and an active state at the compressed bit-rate. Unlike the PSTN, which generally does such silence suppression across trans-continental links, IP telephony performs silence suppression at the end-points. Hence, no network support is required to take advantage of end system silence suppression. This leads to a reduction in cost to perform the silence suppression (as it is distributed to end systems where it can be done cheaply).

2.1.2 Subjective Voice Quality

Each codec provides a certain quality of voice. The quality of transmitted voice is a subjective response of the listener. A common benchmark used to determine the quality of sound produced by specific codecs is the mean opinion score (MOS). With MOS, a wide range of listeners judge the quality of a voice sample (corresponding to a particular codec) on a scale of 1 (bad) to 5 (excellent). Scores are averaged to provide the MOS for that sample. MOS scores and correspondent quality impairment are shown in Table 2.1.

Table 2.1: MOS subjective quality scores and correspondent impairment

Mean Opinion Score (MOS)	Quality	Impairment description
5	Excellent	Imperceptible
4	Good	Just perceptible, not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying but not objectionable
1	Bad	Very annoying and objectionable

It is important to notice that MOS scores define the upper limit for achievable end-to-end quality. This will determine voice quality for perfect network conditions (no packet loss, delay, jitter or other quality-degrading factors). Figure 2.1 illustrates the MOS according to coding techniques and codec bit rate⁴. Although waveform codecs provide high voice quality with coding rates of 32-64 kbits/s, their quality falls rapidly at bit rates of 16 kbits/s and lower. Hybrid codecs provide good quality compared to source codecs operating at the same low bit rates. Quality of hybrid codecs is slightly lower than that of waveform codecs operating at higher bit rates (64 kbits/s).

2.1.3 VoIP Standard Codecs

The commonly used VoIP codecs are G.711 [17], G.729 [18] and G.723.1 [19], which are standardized by the ITU-T in its G-series recommendations. G.711 sample based PCM codec generates 64 kbits/s. In the same family of sample based codecs, G.726 [20] achieves 16-40 kbits/s using ADPCM. Recent frame-based encoders can be used in order to provide drastic rate reduction (e.g., 8 kbits/s for G.729, 5.3 and 6.3 kbits/s for G.723.1).

Typically, VoIP codecs use narrow-band voice sampling at 8 kHz. Increasing the sampling frequency to the 16 kHz used for wide-band voice coding produces much more comfortable, and

⁴Source: A.M. Kondoz, *Digital Speech Coding for Low Bit-Rate Communications Systems*, 1995.

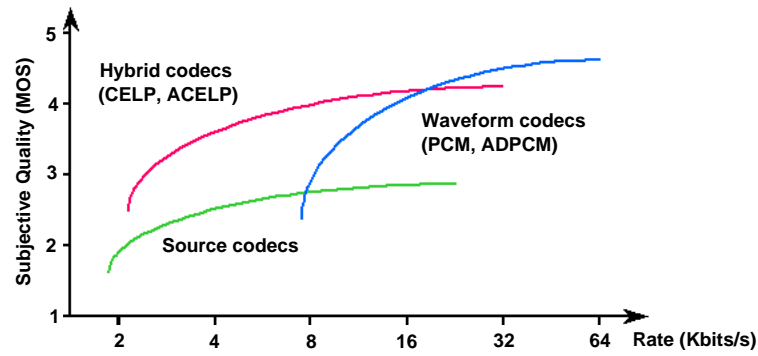


Figure 2.1: Subjective codec quality: effect of coding technique and codec bit rate

intelligible voice [21]. However, wide-band voice coding has found use limitations because of interactions with the Public Switched Telephone Network (PSTN). These limitations do not exist in the case of VoIP communications when the call is initiated and terminated within the IP network. Therefore, because of the dramatic quality improvement, future VoIP codecs will tend to be wide-band.

2.1.3.1 G.711 PCM Standard Codec (64 kbits/s)

Originally designed for circuit switched telephony, G.711 codec [17] uses PCM technique to generate 8 bits samples per 0.125 ms, leading to a data rate of 64 kbits/s. G.711 uses a relatively simple way to digitize analog voice by using semi-logarithmic scale. The goal of this technique is to increase the resolution for small signals, while large signals are treated proportionally. In America μ -law G.711 coding is the standard, while in Europe the slightly different A-law coding is used. Because of their simplicity, excellent quality and low delay, both codecs are still widely used today. G.711 has the advantages of reduced complexity and low delay with high quality reproduced voice, but require a relatively high bit rate and have a high susceptibility to packet loss.

2.1.3.2 G.728 Low Delay CELP Codec (16 kbits/s)

G.728 CELP codec [22], uses backward adaptation for filter coefficients calculation (see hybrid codecs Section 2.1.1.3). But, rather than buffering 20 ms of the input voice to calculate filter coefficients, past reconstructed voice is used to find these coefficients. G.728 uses much shorter frame length than traditional CELP codecs: a frame length of only 5 samples giving it a reduced coding delay of 2.5 ms. At the decoder a post-filter is used to improve the perceptual quality of the reconstructed voice. All this leads to a codec at 16 kbits/s with a delay of 2.5 ms and a good robustness to packet loss.

2.1.3.3 G.723.1 Codec (6.3/5.3 kbits/s)

G.723.1 codec generates low bit rates compared to G.711. Two bit rates are associated with it, 5.3 and 6.3 kbits/s, whose mode of operation can change dynamically at each frame. Its frame length is 30 ms, and a 7.5 ms delay is necessary for its look-ahead buffer, resulting in total coding delay of 37.5 ms. The G.723.1 codec uses linear predictive Analysis-By-Synthesis (ABS) coding technique. The excitation for the high rate is multi-pulse maximum likelihood quantization (MP-MLQ), whereas the low rate codec is Algebraic-Code-Excited Linear Prediction (ACELP). G.723.1 Annex A [23] specifies an improved version of G.723.1 codec that can provide silence suppression (VAD, DTX) and comfort noise generation (CNG).

2.1.3.4 G.729 Codec (8 kbits/s)

The G.729/G.729A codec uses conjugate-structure, algebraic-code-excited linear prediction (CS-ACELP) coding technique. It generates 8 kbits/s rate. The frame length is 10 ms and the look-ahead time is 5 ms, the algorithmic delay is then 15 ms. G.729A [24] is a reduced-complexity optimized version of the original G.729. G.729B codec [25] can provide silence suppression using VAD, DTX and CNG mechanisms.

2.1.3.5 GSM Codec (13 kbits/s)

The “Global System for Mobile communications” (GSM)⁵, is a digital mobile radio system which is extensively used throughout Europe, and also in many other parts of the world. The GSM voice codec [26] operates at 13 kbits/s and uses a Regular Pulse Excited (RPE) coding technique. Basically the input voice is split up into 20 ms long frames, and for each frame a set of 8 predictor coefficients are found. At the decoder, the reconstructed excitation signal is fed through synthesis filters to give the reconstructed voice. A post-filter is used to improve the perceptual quality of this reconstructed voice. The GSM main advantage over other low rate codecs is its relative simplicity.

2.1.3.6 Summary of Codec Features

Features of main standard VoIP codecs [16][27] are summarized in Table 2.2. The coding delay of G.729 and G.723.1 codecs includes, in addition to frame length, 5 ms and 7.5 ms of look-ahead delay, respectively. For the remainder of codecs it is formed only by the frame length. The complexity of a codec is estimated from its Digital Signal Processing (DSP) cost expressed in MIPS (Millions of Instructions Per Second).

As can be shown from this table, there is always a trade-off between the bit rate, the delay, the complexity of a given codec and the quality of its reconstructed voice. For instance, the reduced bandwidth utilization of recent codecs (i.e., G.723.1 and G.729) is at the expense of additional complexity and coding delay (37.5 ms for G.723.1) as well as slightly lower quality compared to

⁵GSM is standardized by ETSI (European Telecommunications Standards Institute)
<http://www.etsi.org>

Table 2.2: Features of main standard VoIP codecs

Codec standard	Compression method	Bit rate (kbits/s)	Coding delay (ms)	Quality (MOS)	DSP cost (MIPS)
G.711	PCM	64	0.125	4.4	0.1
G.726	ADPCM	16/24/32/40	0.125	2/3.2/3.85/4	7
G.728	LD-CELP	16	2.5	3.61	30
G.729	CS-ACELP	8	15	4	30
G.729A	CS-ACELP	8	15	3.7	15
G.723.1	MP-MLQ	6.3	37.5	3.9	16
G.723.1	ACELP	5.3	37.5	3.65	18
GSM	RPE-LTP	13	20	3.71	15

G.711 codec. We can observe that complexity generally increases with decreasing bit rate. Although, MOS would decrease with bit rate in most cases, G.729A has a lower MOS than the 6.3 kbits/s G.723.1. This can be explained by the fact that MP-MLQ coding better reproduces voice than ACELP one. The industry trend appears to be developing codecs that utilize less bandwidth than their predecessors (G.723.1 utilizes less than one-twelfth of G.711 bandwidth). Although it might seem logical from a resource usage viewpoint to convert all VoIP calls to low bit-rate codecs to save on infrastructure costs, there are drawbacks to compressing voice. One of the main drawbacks is signal distortion due to multiple encodings (tandem encodings). For example, when a G.729 voice signal is tandem-encoded three times, the MOS score drops from 4 to 2.68 [21].

2.2 Transport of Real-time Voice over the Internet

After the coding operation, a host wishing to send media packets concatenates a certain number of voice samples (for G.711) or a certain number of frames (e.g., G.729 and G.723.1 frames) into packets of equal sizes, and adds protocol headers. The protocol stack used to carry the real time voice packets is RTP over UDP/IP. Voice packets are encapsulated with RTP, UDP, and IP headers then sent into the network, either to a multicast group or unicast to another participant. (Figure 2.2). At the receivers side, data is de-packetized and forwarded to a playout buffer, which smooths out the delay jitter incurred by network transmission. The voice data is decoded and the reconstructed voice signal is delivered to the listener.

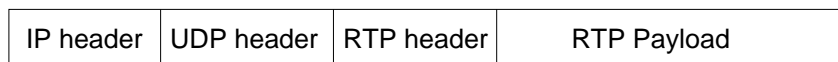


Figure 2.2: RTP packet encapsulation

Packet switched networks use the UDP/IP protocol stack to transport real-time multimedia traffic. IP provides connectionless, unreliable, best-effort packet delivery service. The IP protocol [28] implements two basic functions namely addressing and fragmentation. IP receives segments from

the upper host-to-host layer (i.e., application layer) and fragments them into datagrams (packets). These packets are transmitted from sources to destinations and then reassembled back into segments on the receiving side IP layer. Each IP packet contains IP addresses of the sender and receiver and is treated independently at intermediate nodes. Routing algorithms are applied on a packet-per-packet basis to make next-hop decisions. In IP best-effort networks, the best path for a given destination is determined from hop-by-hop decisions based on routing tables. UDP (User Datagram Protocol) [29] is the transport layer protocol used jointly with IP protocol. UDP is a connectionless unreliable protocol that neither creates a virtual circuit nor contacts the destination before starting delivering information to it. It does not guarantee delivery of packets nor duplicate protection (it neither checks if packets are received nor acknowledges packets receipt). UDP assumes that reliability will be provided at the application layer. The simplicity of UDP affords minimal overhead needed by real-time applications. In addition to media transport protocol, the use of a signaling protocol to set up the call connection and negotiate the media format that will be used during the communication is required.

2.2.1 RTP for Voice Media Transport

Real-time flows such as voice and video streams have a number of common requirements that distinguish them from traditional Internet data services [30]:

Sequencing: packets must be re-ordered in real time at the receiver, in case they arrive out of order.

If a packet is lost, it must be detected and compensated for without retransmissions.

Intra-media synchronization: the amount of time between when successive packets are to be played out must be conveyed. For example, no data is usually sent during silence periods in speech. The duration of this silence must be reconstructed properly.

Inter-media synchronization: if a number of different media are being used in a session, there must be a means to synchronize them, so that the audio that is played out matches the video.

Payload identification: in the Internet, it is often necessary to change the encoding for the media (payload) on the fly to adjust to changing bandwidth availability or the capabilities of new users joining a multicast group. Some kind of mechanism is therefore needed to identify the encoding for each packet.

Frame indication: video and audio are sent in logical units called frames. It is necessary to indicate to a receiver where the beginning or end of a frame is, in order to aid in synchronized delivery to higher layers.

The Real-time Transport Protocol (RTP/RTCP) [8][9] provides end-to-end network transport functions required by applications transmitting real-time data (e.g., VoIP, video). RTP was designed to be independent of encoding schemes, network layers and signaling protocols. It is intended to be malleable to provide the information required by a particular application, and is usually integrated

into the application rather than being implemented as a separate layer. In addition, RTP can be tailored through modifications and/or additions to the protocol headers. However, RTP does not intrinsically provide any mechanism to guarantee QoS or resource reservation for real-time services.

Applications typically run RTP on top of UDP to make use of its multiplexing and checksum services. In fact, both protocols contribute parts of the transport protocol functionality. Basically, RTP provides payload type identification, sequencing, and time-stamping. Nevertheless, RTP does not ensure timely delivery or prevent out-of order delivery, at the same time it does not assume that the underlying network is reliable and delivers packets in sequence.

It is important to notice that RTP provides some functionality beyond basic functionalities such as re-sequencing and loss detection:

Multicast-friendly: RTP/RTCP have been engineered for multicast. In fact, they are designed to operate in both small multicast groups, like those used in a three-person phone call, to huge ones, like those used for broadcast events.

Media independent: RTP provides services needed for generic real-time media, such as voice and video. Any codec-specific additional header fields and semantics are defined for each media codec in separate specifications.

Encryption: RTP media streams can be encrypted using keys that are exchanged by some non-RTP method, e.g., Session Initiation Protocol (SIP) [31] or the Session Description Protocol (SDP) [32].

2.2.1.1 RTP Packet Format

The RTP packet consists of a fixed RTP header (12 bytes), a list of contributing sources (may be empty) and payload data. Encapsulation of the RTP packet may be needed by the underlying protocols. Normally a packet of the underlying protocol has a single RTP packet [38]. In the following, we describe the RTP header fields.

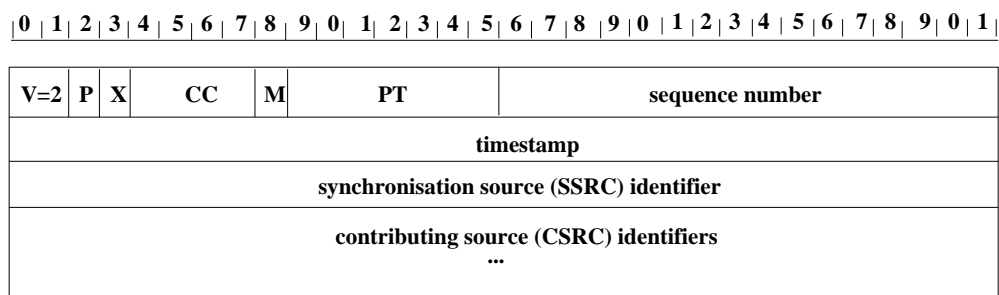


Figure 2.3: RTP header format

The RTP header (Figure 2.3) is 12 bytes long. The field *V* identifies the version of the RTP. If the padding bit *P* is set, the payload is padded to ensure proper alignment. The last byte of the padding has a count of how many padding bytes should be ignored. Padding may be needed for

carrying several RTP packets in a lower-layer protocol data unit or for some encryption algorithms with fixed block sizes.

The *X* flag signals the presence of a header extension between the fixed header and the payload. The extension mechanism allows experimentation with new payload-format-independent functions that require additional information to be carried in the RTP data packet header.

RTP supports the notion of media-dependent framing to assist in the reconstruction and playout process. The marker *M* bit provides information for this purpose. For audio, the first packet in a voice talk-spurt can be scheduled for playout independently of those in the previous talk-spurt. The bit is used in this case to indicate the first packet in a talk-spurt. For video, a video frame can only be rendered when its last packet has arrived. Here, the marker bit is used to indicate the last packet in a video frame (frame boundaries in the packet stream).

The Payload Type *PT* field identifies the media encoding used in the packet. A profile specification document defines a set of payload type codes and their mapping to payload formats. This field indicates, for instance, what type of media encoding (such as PCM, ADPCM or LPC) is contained in each packet so that senders can change the encoding during a conference, for example, to accommodate a new participant that is connected through a low-bandwidth link or react to indications of network congestion [8].

Users within a multicast group are distinguished by a random 32 – bit synchronization source *SSRC* identifier. Having an application-layer identifier allows to easily distinguish streams coming from the same translator or mixer (see section 2.2.2 below) and associate receiver reports with sources. This identifier is chosen randomly so that no two synchronization sources within the same RTP session will have the same *SSRC* identifier. In the rare event that two users happen to choose the same identifier, they redraw their *SSRC*s.

As described below, a mixer combines media streams from several sources, e.g., a conference bridge mixes the audio of all active participants. The contributing source list (*CSRC*) identifies all the contributing sources for the payload contained in this packet in order to help the participants to distinguish who happens to be speaking at any given time. For an audio conference, it would list all active speakers. The *CSRC* count (*CC*) field indicates the number of *CSRC* identifiers that follow the fixed header.

The *sequence number* which increments by one for each RTP data packet is used by the receiver to detect packet loss and for restoration of packet sequence. A random value is chosen for the start of the sequence number to make known plain-text attacks on encryption more difficult.

The information pertaining to the sampling instant of the first byte in the RTP data packet is contained in the *timestamp* field. The sampling instant is derived from a clock that increments monotonically and linearly in time allowing synchronization and jitter calculations. The resolution of the clock must be sufficient for the desired synchronization accuracy and for measuring packet arrival jitter. Time is represented in seconds using the timestamp format of the Network Time Protocol (NTP) [39]. As in the case of sequence numbers the initial value of the timestamp is random.

2.2.2 RTP Mixers and Translators

RTP mixers are devices which take media from several users, mix or bridge them into one media stream, and send the resulting stream out. Mixers limit the bandwidth if several sources send data simultaneously, fulfilling the function of conference bridge. This is especially useful in the case where multicast session participants in one area are connected through a low-bandwidth link to the majority of participants who have high-bandwidth network access. Instead of forcing RTP sources to use a lower bit rate, and thus reduced-quality audio encoding, an RTP-level relay (mixer) may be placed near the low-bandwidth area. This mixer resynchronizes incoming audio packets to reconstruct the constant spacing interval generated by the sender (i.e., 20 ms), mixes these reconstructed audio streams into a single stream. This allows receivers that are connected via fast links to still receive high quality media.

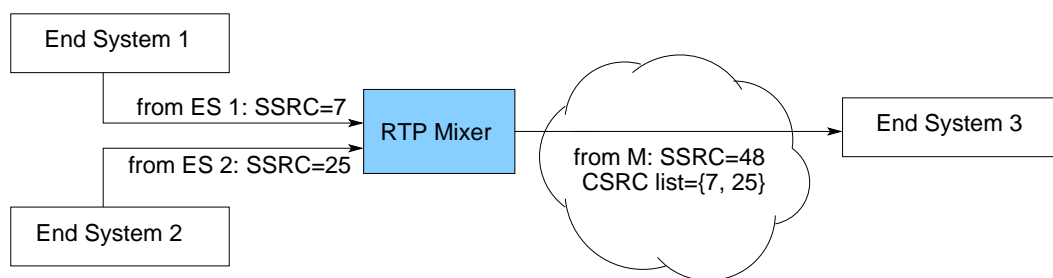


Figure 2.4: RTP mixer

In addition, RTP translators can be used to further reduce the bandwidth requirements of a stream by converting it to a lower-bandwidth format before sending it over a low-bandwidth link. Only in the case where a mixer is involved, the list of contributing source identifiers CSRC is present in the RTP header (Figure 2.4).

The RTP header includes a means for mixers to identify the sources that contributed to a mixed packet so that correct talker indication can be provided at the receivers [8]. RTP includes explicit support for mixers and translators.

2.2.3 RTCP for Media Transmission Control and Management

RTCP (Real-time Transport Control Protocol) is the companion control protocol for RTP. It is based on the periodic transmission of control packets to session participants, using the same distribution mechanism as the data packets (Figure 2.5).

The underlying protocol must provide multiplexing of the data and control packets, for example, using different port numbers with UDP. On this note RTP must be assigned an even UDP port number and the corresponding RTCP is assigned the next higher (odd) UDP port number. RTCP performs the basic following functions:

- Provides feedback on the quality of the data distribution (loss rate, delay, jitter)

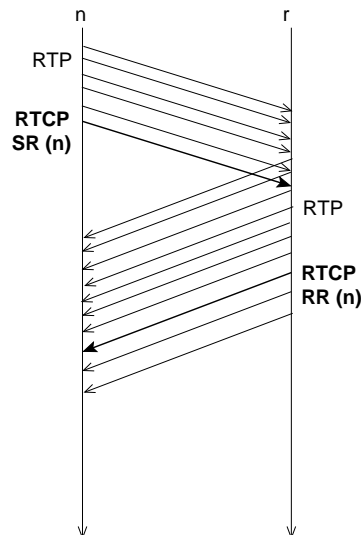


Figure 2.5: RTCP sender and receiver reports sent conjointly with RTP packets

- Carries a persistent transport-layer identifier for an RTP source (canonical name)
- Controls the rate in order for RTP to scale up to a large number of participants (e.g., audio/video conference)

These functions are mandatory when RTP is used in a multicast environment, and are recommended for all environments. They are exhaustively performed using the five types of RTCP packets: Sender Report (SR), Receiver Report (RR), Source Description (SDS), end of participation to a session (BYE) and application-specific packets (APP). Each RTCP packet contains a fixed size header followed by structural elements: source report blocks. Different RTCP packet types can be stacked together to form a compound RTCP packet. Figure 2.6 and Figure 2.7 show the format of RTCP Receiver Report and Sender Report, respectively.

In a multicast environment, media senders and receivers periodically send RTCP packets to the same multicast group (but different ports) as is used to distribute RTP packets. Each RTCP packet contains a number of elements, usually a SR or a RR followed by source descriptions (SDS). Sender Reports are generated by users who are also sending RTP media. They describe the amount of data sent so far, as well as correlating the RTP sampling timestamp and absolute time to allow synchronization between different media. Receiver Reports are sent by RTP session participants which are receiving media.

RTCP RR allow receivers to provide a feedback to all members of a group on the quality of the reception. RTP sources may use this information to adjust their data rate, while other receivers can determine whether QoS problems are local or network-wide. External observers can use it for scalable QoS management.

Source Description (SDS) packets are used for session control. They contain the *CNAME* (Canonical Name), a globally unique identifier similar in format to an email address. The *CNAME* is used for resolving conflicts in the SSRC value and associate different media streams generated

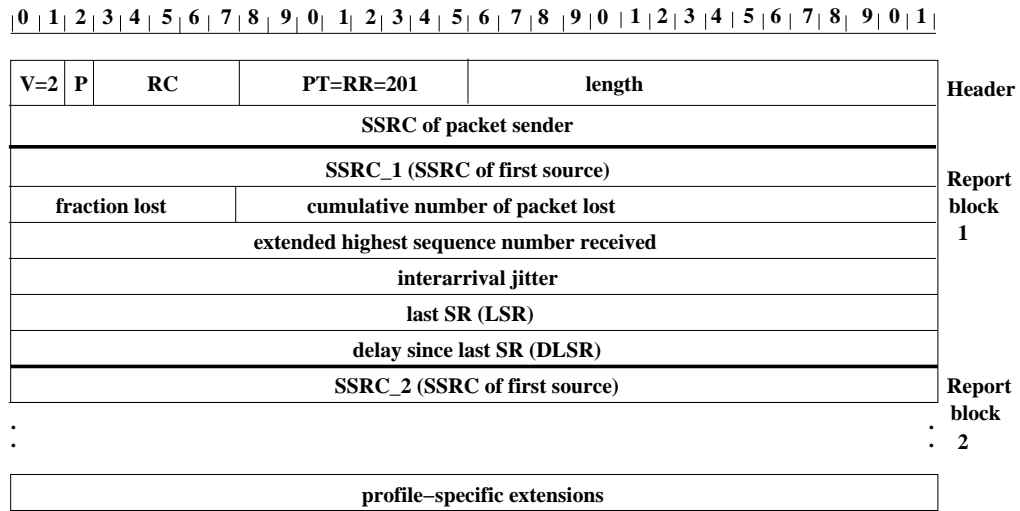


Figure 2.6: Packet format of the RTCP Receiver Report

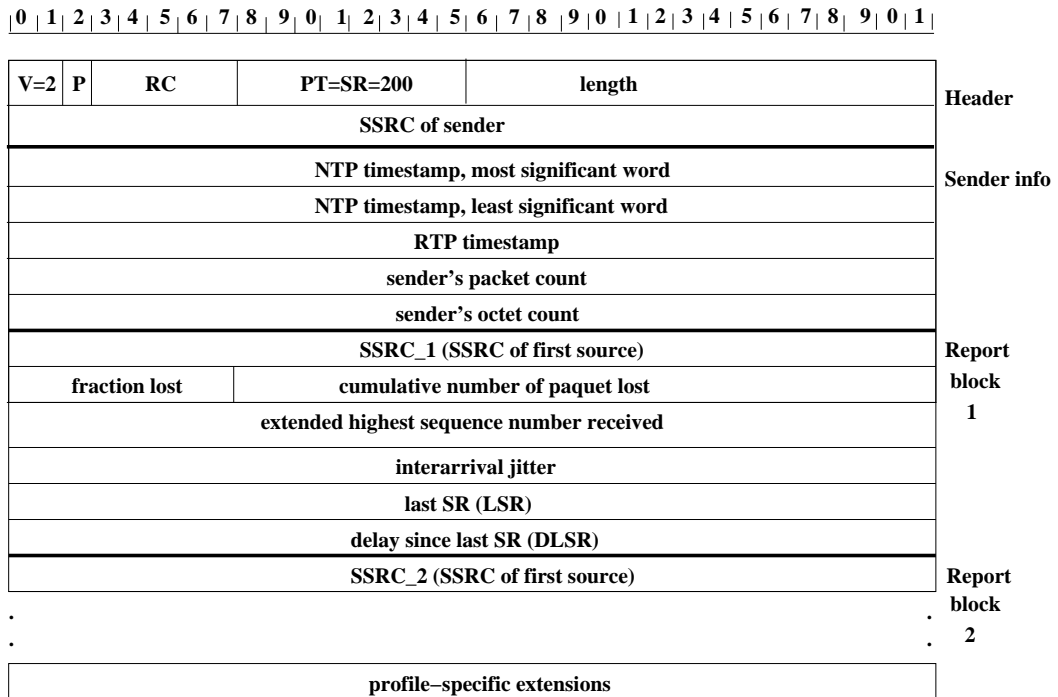


Figure 2.7: Packet format of the RTCP Sender Report

by the same user. SDES packets also identify the participant through its name, email, and phone number. This provides a simple form of session control. Client applications can display the name and email information in the user interface. This allows session participants to learn about the other participants in the session. It also allows them to obtain contact information (such as email and phone). If a user is leaving, he includes a BYE message. Finally, application (APP) elements can be used to add application-specific information to RTCP packets. Since the sender reports, receiver reports, and SDES packets contain information which can continually change, it is necessary to send these packets periodically [30].

RTP specification [8] recommends that the collective RTCP feedback bandwidth must not exceed 5% of data bandwidth. Of this RTCP feedback bandwidth, 25% (i.e., 1.25% of RTP data bandwidth) is allocated to sender reports and 75% (i.e., 3.75% of RTP data bandwidth) is allocated to receiver reports.

2.2.3.1 RTCP Feedback Statistics

Each receiver report contains one block for each RTP source in the group. Each block describes the *instantaneous* and *cumulative loss rate* and *jitter* from that source. The block also indicates the *last timestamp* and *delay since receiving a sender report*, allowing sources to estimate their distance (in terms of delay) to receivers.

Round-Trip Time Estimation

An RTP Source $SSRC_n$ can estimate the round-trip propagation delay to a receiver $SSRC_r$ by recording the time A when this reception report block is received (Figure 2.8). It calculates the total round-trip time $A - LSR$ using the last SR timestamp (LSR) field, and then subtracting this field to leave the round-trip delay as:

$$RTT = A - LSR - DLSR \quad (2.2)$$

Interarrival Jitter

The interarrival jitter field contained in RTCP reports, represents an estimate of the statistical variance of the RTP data packet interarrival time, measured in timestamp units. The interarrival jitter J is defined to be the mean deviation (smoothed absolute value) of the difference D in packet spacing at the receiver compared to the sender for a pair of packets. As shown in the Equation 2.3, this is equivalent to the difference in the transmission delay for two packets. This delay is the difference between a packet's RTP timestamp and the receiver's clock at the time of arrival, measured in the same units. If S_i is the RTP timestamp from packet i , and R_i is the time of arrival in RTP timestamp units for packet i , then for two packets i and j , D may be expressed as:

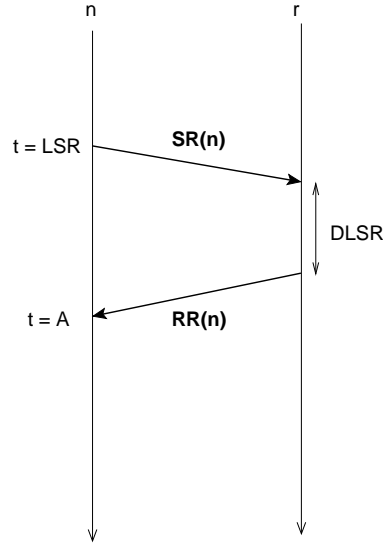


Figure 2.8: RTT estimation

$$D(i, j) = (R_j - R_i) - (S_j - S_i) = (R_j - S_j) - (R_i - S_i) \quad (2.3)$$

The interarrival jitter is calculated continuously as each data packet i is received from source $SSRC_n$ using this difference D for that packet and the previous packet $(i-1)$ in order of arrival (not necessarily in sequence), according to the Formula 2.4:

$$J(i) = J(i-1) + (|D(i-1, i)| - J(i-1))/16 \quad (2.4)$$

Whenever a reception report is issued, the current value of J is sampled. This algorithm is the optimal first-order estimator and the gain parameter $1/16$ gives a good noise reduction ratio while maintaining a reasonable rate of convergence [8].

2.2.3.2 RTCP Scale Issue in Case of Multicast Transmission

In the case of a multicast session, if the RTP participants simply sent RTCP packets with a fixed period, the resulting bandwidth used in the multicast group would grow linearly with the group size. This is clearly undesirable. Instead, the period between RTCP packets from each user is then set to scale linearly with the number of group members. Each session member estimates the number of other session members it hears from (via RTCP packets). This ensures that the bandwidth used for RTCP reports remains fixed, independently of the group size. However, since the group size estimate is obtained by counting the number of other participants, it takes time for each new participant to converge to the correct group size count [30].

2.3 Network Impairments and Proposed Improvements

The best-effort delivery service offered by the Internet does not ensure timely delivery of packets and can result in highly variable packet delays, jitter and loss. In the following we analyze the main network impairments that could encounter VoIP traffic and the main improvements proposed in the literature.

2.3.1 End-to-End Delay

VoIP packets sent over a best-effort network, are subject to variable delays. In fact, the transmission delay of packets, the time needed to transmit a packet from the sender to the receiver, is variable and depends on the current network conditions and the routing path [40][41][42]. In addition to the transmission delay, end-to-end delay depends on the voice codec, and the payload size of the packets. The end-to-end delay components can be summarized in:

- Coding/decoding delay (including look-ahead, processing and packetization delay).
- Propagation delay: speed of light delay through the network path. This delay is very small, especially for high speed links.
- Transmission delay: the time required to receive the entire packet before processing and forwarding it through the router. This delay is determined by the packet length and transmission speed. Using short packets over high-speed trunks can easily shorten the transmission delay but potentially decrease network efficiency.
- Switching/Routing delay: this delay is the time the router takes to switch packets. This time is needed to analyze the packet header, check the routing table, and route the packet to the output port. This delay depends on the architecture of the route engine and the size of the routing table.
- Queuing delay: introduced in router buffers due to the statistical multiplexing nature of IP networks, this delay is a function of the traffic load on a router, the length and the statistical distribution of packets. Designing very large router and link capacities can reduce but not completely eliminate this delay. Queuing delay can be divided into:
 - self queuing delay (queuing behind previous packets of the same flow)
 - queuing delay due to cross traffic

No matter how well VoIP devices and networks are designed, a fundamental delay exists that simply cannot be eliminated. That is, some delay will always be introduced as a result of the physical limits of coding, packetization, processing time, routing and queuing delays. More precisely, we should note that the IP network delay, primarily determined by the buffering, queuing, and switching or routing delay of IP routers, represents the most important part of VoIP delays.

It is recognized that the end-to-end delay has a great impact on the perceived quality of interactive conversations with a threshold effect around 150 ms one-way delay according to ITU-T recommendation G.114 [3]. Delays below 150 ms should not affect interactivity. Nevertheless, the impact of delay on voice communication quality varies significantly with the use. For instance, long delays are not annoying in a cell phone as in a regular wired phone because of the added value of mobility. Table 2.3 shows tolerable delay limits for main standard codecs in the absence of packet loss (according to ITU-T recommendation G.113).

Table 2.3: Tolerable delay limits in the absence of packet loss

ITU-T standard codec	Bit rate (Kbps)	Maximum delay (ms)
G.711	64	400
G.729A	8	296
G.723.1	5.3	221
G.723.1	6.3	253

The IETF developed the Integrated Services (IntServ) network architecture [5] to provide individualized quality of service guarantees to real-time multimedia applications with stringent delay requirements. IntServ defines three major classes of service: the Guaranteed-Service (GS) [43] class, which provides for delay-bounded service agreements; the Controlled-Load (CL) [44] service class, which provides a form of statistical delay service agreements (nominal mean delay) and the well-known best-effort service. The RSVP protocol [6] is used for signaling in the former two service classes. Sender and receiver periodically send RSVP messages (reservations are maintained in a soft state) and these messages make the routers to reserve resources. The support of per flow guarantees poses severe scalability problems. RSVP is useful for access networks, but demands high capacity in the core as the number of flows grows [45]. In order to leverage that scalability problem, the Differentiated Services (DiffServ) architecture [7] was developed within the IETF.

With differentiated services, there are two important router functions that need to exist. First, the router closest to the sending application (the “ingress router”) needs to recognize the packets that should get better service and mark them. All other packets should be marked to indicate that they are not getting better service. Second, all other routers need to treat the marked packets properly so that they receive the service they asked for, even in times of network congestion. Note that the interior routers do not know about individual network flows, but work with aggregates of flows based on the marking. This architecture simplifies the operation of the intermediate nodes, moving the complexity to the edge nodes. DiffServ may be used in conjunction with IntServ applied in different parts of a network; the former in the core and the latter in the edge. Nevertheless, most routers in the current Internet employ FIFO (First In First Out) queue management coupled with a drop-tail buffer management policy. None of these network QoS mechanisms have yet seen widespread deployment, mainly because they require modifications in the current Internet network architecture.

2.3.2 Jitter

Jitter is variation in the delay of arrivals of voice packets at the receiver caused mainly by fluctuations in queuing delay. Jitter introduces a discontinuity of the voice stream and is usually compensated for by using a playout buffer to play out uninterrupted voice. Playout control can be exercised both in adaptive or static playout delay mode. In static mode, the playout delay d_{playout} is chosen close to the maximum of the variable part of the end-to-end delay. Playout buffer size is usually configured as twice the number of packets that can be generated during d_{playout} to avoid buffer overflow. The value of d_{playout} can be obtained, in this case, from call control entity (e.g., SIP proxy) as configuration of the VoIP receiver. Although a static method, using fixed playout buffer, is easier to implement than an adaptive method, it can cause unsatisfactory audio transmission quality because there is no optimal delay value when network conditions vary with time.

In adaptive playout mode, the receiver dynamically determines the playout buffer size, at the beginning of each talk-spurt, according to the change in the buffer occupancy level and the observed network delay in order to avoid buffer overflow even in case of early packet arrival. The playout adjustment is performed during the silent periods between talk-spurts. To determine d_{playout} value, the following tradeoff should be considered at the receiver. If d_{playout} is too small, buffer underflow can occur when packets arrive late and this will be considered as a late arrival packet loss. Although, large d_{playout} value will decrease the probability of buffer underflow, this will increase the overall end-to-end delay.

Since the current network delay characteristics are not known in advance, adaptive playout algorithms calculate time of each incoming talk-spurt based on the delays experienced by already-received packets. The first algorithms were proposed in [10]. The basic idea is the following:

Let n_i be the total delay of audio packet i introduced by the network. Estimation of the average network delay \hat{d}_i and the average delay variation \hat{v}_i is calculated for each incoming packet:

$$\hat{d}_i = A * \hat{d}_{i-1} + (1 - A) * n_i \quad (2.5)$$

$$\hat{v}_i = A * \hat{v}_{i-1} + (1 - A) * |\hat{d}_i - n_i| \quad (2.6)$$

A is a constant weighting factor that characterizes the memory properties of this estimation, it is used in order to limit sensitivity to short-term packet jitter. These estimations are recomputed each time a packet arrives, but used only when a new talk-spurt is initiated. Most recent values are then used to calculate the playout delay of the first packet in the talk-spurt using Equation 2.7:

$$d_{\text{playout}}^i = \hat{d}_i + B * \hat{v}_i \quad (2.7)$$

Subsequent packets of that talk-spurt are played out with rate equal to the generation rate at the sender (i.e., codec rate). Constant B is usually set to 4. The larger this coefficient is, the more packets are played out at the expense of longer delays.

2.3.3 Packet Loss

In the Internet, packets may get lost due to network congestion or wireless transmission errors. Tolerable loss rates depend heavily on the voice codec [46], and are in the range of 1 to 10% [47]. At the receiver, playout buffer delivers a continuous stream of packets to the depacketizer and eventually to the decoder which reconstructs the voice signal. Decoders often implement Packet Loss Concealment (PLC) that produces a replacement for a lost packet, similar to the original one, by filling in silence or noise, by interpolating or even by regenerating the packet from the surrounding ones.

A method for error concealment based on extrapolating/interpolating received voice segments is described in Appendix I to G.711. However, this is an heuristic procedure and does not guarantee robust operation. Error concealment works best for small loss rates and durations [42]. Table 2.4 illustrates tolerable loss rates of main standard codecs for delay values under 150 ms (according to ITU-T recommendation G.113).

Table 2.4: Tolerable packet loss rates for delay values less than 150 ms

ITU-T standard codec	Bit rate (Kbps)	Maximum packet loss rate (%)
G.711 without PLC	64	1
G.711 with PLC	64	10
G.729A with VAD	8	3.4
G.723 with VAD	6.3	2.1

Another approach which represents one of the primary mechanisms used for end-to-end packet loss recovery over packet switched networks and also for quality improvement of noisy transmission over wireless links is Forward Error Correction (FEC). The idea behind FEC is to transmit jointly with the original data a redundant data flow so that a corrupted packet can be reconstructed at the receiver. The main drawback of FEC technique is the packet delay increase due the waiting time to play the reconstruction of a lost packet from next packets.

In networks where bandwidth limitations are in packets/sec, FEC does not require additional bandwidth since FEC redundant information is transmitted jointly in packets of the original flow. However, in today's Internet, bandwidth limitations are in bytes/sec rather than in packets/sec and consequently FEC will need additional bandwidth which decreases transmission efficiency and increases network level of congestion. The amount of redundant information should be adaptive to avoid taking bandwidth away from other flows sharing the same network links.

In [11], Bolot et al. proposed a joined rate and FEC control for audio packet transmission. Based on packet loss feedback information obtained from the receiver, the sender lowers coding rate and/or increases the amount of FEC. In [12], the authors optimized a joint FEC and playout delay adjustment method to tackle the problem of the delay introduced by FEC. The goal of this elaborate method is to analytically find the best coding rate, the amount of FEC and the playout

delay as a function of network characteristics so as to maximize the perceived audio quality. The playout delay is adapted separately and the best FEC scheme is chosen given this playout delay.

2.3.4 Echo Impairment

An important voice quality impairment is the echo generated from the reflection of a call participants signals, perceived as delayed and attenuated versions of their own voices. The larger the end-to-end delay, the more annoying is the echo. In addition, the presence of echo has a significant effect on sensitivity to delay. If the echo exceeds approximately 25 ms, it can be distracting and cause breaks in the conversation. Although one might at first think that echo cannot happen in a packetized voice system, reflections may indeed happen (i) in the case of a communication between a circuit switched network (PSTN) and IP packet networks because of impedance mismatch from the four-wire network switch conversion to the two-wire local loop, and (ii) because of acoustic feedback at the PC end-point when the microphone picks up the remote persons voice from the speaker as well as multiple reflections in the room and bounces them back [42]. Both types of echo can be controlled by an Echo Canceler, that should be located as close to the source of echo as possible. In voice over packet-based networks, echo cancelers are built into the low bit-rate codecs and are operated on each DSP. Echo cancellation algorithms can remove most of the echo effects as long as the latency is not too high.

2.4 The E-Model for Real-Time Audio Quality Measurement

Perceived voice quality is typically measured by the Mean Opinion Score (MOS) presented in Section 2.1.2. MOS subjective quality score ranges from 1 (unacceptable) to 5 (excellent) and can be obtained by carrying out subjective tests. Some objective quality models, can also be used to generate scores by comparing the impaired voice signal with its original version as in PESQ [48]. However, PESQ do not consider the effect of delay on voice communications and neither MOS nor PESQ can be used for real-time on-line quality estimation.

The ITU-T *E-model* provides a framework for real-time on-line quality estimation from network performance measurements (e.g., delay and loss characteristics) and application level factors (e.g., low bit rate codecs). It represents an analytic model of voice quality defined in the ITU-T recommendation G.107 [49] as well as other associated ITU-T recommendations [50][51][52]. The result of the E-model is the calculation of the *R-factor* (ranging from a best case of 100 to a worst case of 0). The R-factor can be further translated into MOS scale through these expressions:

$$\text{MOS} = \begin{cases} 1 & \text{for } R < 0 \\ 1 + 0.035R + 7 \cdot 10^{-6}R(R - 60)(100 - R) & \text{for } 0 < R < 100 \\ 4.5 & \text{for } R > 100 \end{cases} \quad (2.8)$$

Equation 2.8 is plotted in Figure 2.9.

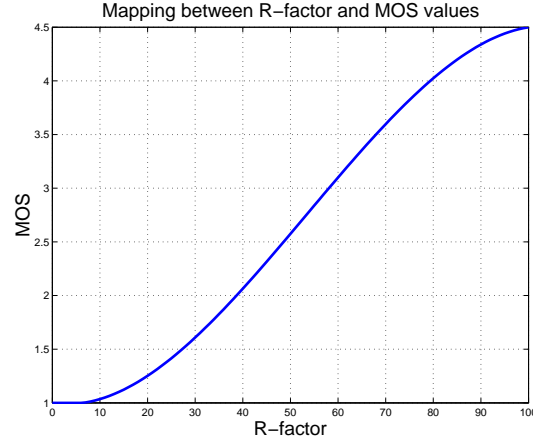


Figure 2.9: R-factor and MOS mapping

The R-factor is defined in [49] as:

$$R = R_0 - I_s - I_d - I_e + A \quad (2.9)$$

R_0 represents the basic signal-to-noise ratio.

I_s reflects the impairments occurring simultaneously with the voice signal due to quantization, it is a function of several parameters, none of which is related to the underlying packet transport.

I_d models the impairments caused by one-way delay. Voice quality degrades more rapidly when this delay exceeds 177.3 ms. This effect is modeled using Equation 2.10 [49]:

$$I_d = 0.024 d + 0.11 (d - 177.3) H(d - 177.3) \quad (2.10)$$

where d is the one-way delay (in milliseconds) and $H(x) = 0$ if $x < 0$ and 1 if $x \geq 0$.

I_e is the equipment impairment factor that covers the distortion of the original voice signal due to low-rate codec and packet loss in both the network and the playout buffer. Table 2.5 shows intrinsic I_e values in the absence of packet loss according to Appendix I/G.113[53]. It is important to notice that recent encoding schemes such as G.723.1 and G.729 achieve higher compression rates at the expense of lower intrinsic quality compared to G.711. This makes them less tolerable to loss during their transmission. I_e depends also on whether or not PLC is used [42].

These intrinsic I_e values are not related to other input parameters such as packet loss rate. To take into consideration packet loss, the effective equipment Impairment factor $I_{e\text{eff}}$, which is packet-loss dependent, is derived using the codec specific value for the equipment Impairment factor at zero packet-loss I_e and the packet-loss robustness factor B_{pl} . The packet-loss robustness

Table 2.5: Intrinsic equipment Impairment factor I_e

Codec	Bit rate (kbits/s)	Equipment impairment factor I_e
G.711	64	0
G.726	40	2
G.726	32	7
G.726	24	25
G.726	16	50
G.728	16	7
G.723.1	5.3	19
G.723.1	6.3	15
G.729	8	10
G.729 AB	8	11

factor Bpl is defined as codec specific value. Some Bpl values are listed in Appendix I/G.113 [53] for several codecs (Table 2.6).

Table 2.6: Packet-loss robustness factor Bpl

Codec	Bit rate (kbits/s)	Packet-loss robustness factor Bpl
G.711	64	4.3
G.711 with PLC	64	25.1
G.723.1	6.3	16.1
G.729 AB	8	19.0

These values are obtained from subjective mean opinion score test results as well as network experiences. With the packet-loss probability Ppl, I_{eff} is calculated using the formula 2.11:

$$I_{eff} = I_e + (95 - I_e) * \frac{Ppl}{Ppl + Bpl} \quad (2.11)$$

As can be seen from this formula, the effective equipment Impairment factor I_{eff} in case of $Ppl = 0$ (no packet loss) is equal to the intrinsic I_e value.

Figure 2.10 illustrates the quality degradation given by MOS quality values⁶ decrease when intrinsic I_e values and packet loss rate are increased.

A the advantage factor A represents the deterioration that callers accept to tolerate in return for the access advantage (e.g., mobile phone). For the purpose of comparison to PSTN calls, this factor is usually set to 0.

Typically only I_e and I_d are considered variable in a VoIP system, default values recommended

⁶These values are obtained from the conversion of E-Model values using equation 2.8.

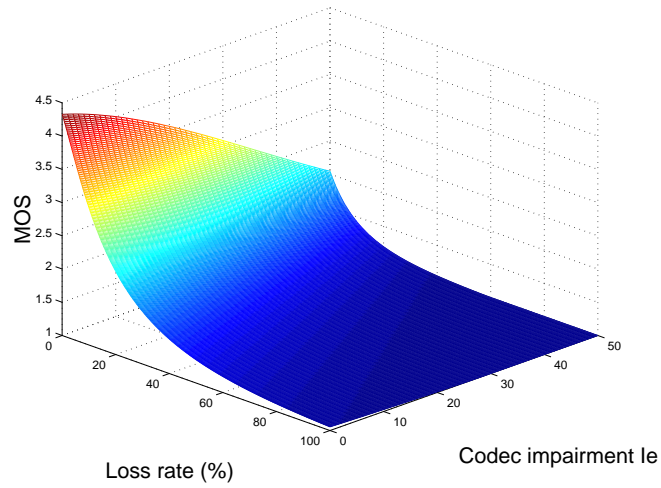


Figure 2.10: MOS vs. intrinsic codec Impairment factor I_e and loss rate

in [49] are used for the remainder of factors in Equation 2.9. The model can then be reduced to Equation 2.12:

$$R = 94.2 - I_e - I_d \quad (2.12)$$

Modeling I_e and I_d as additive factors does not imply that delay and packet loss are uncorrelated, but only that their contributions to the estimated impairments are separable [54]. We illustrate in Figures 2.11 and 2.12 the MOS score of main standard VoIP codecs as a function of both one-way delay and loss rates.

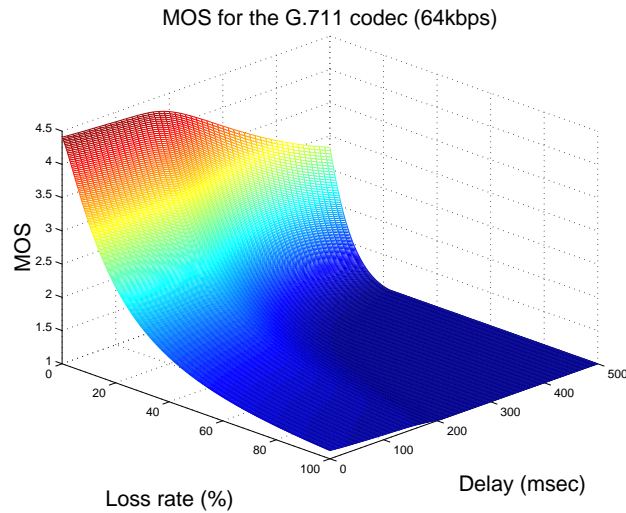


Figure 2.11: MOS of the G.711 codec as a function of network delay and mean loss rate

We observe that MOS score relates to delay and loss in a non-linear fashion. The same delay affects voice quality differently for different values of loss rates, and vice versa. Even if good quality can be achieved for one-way delay values (<150 ms), losses have a more significant impact on

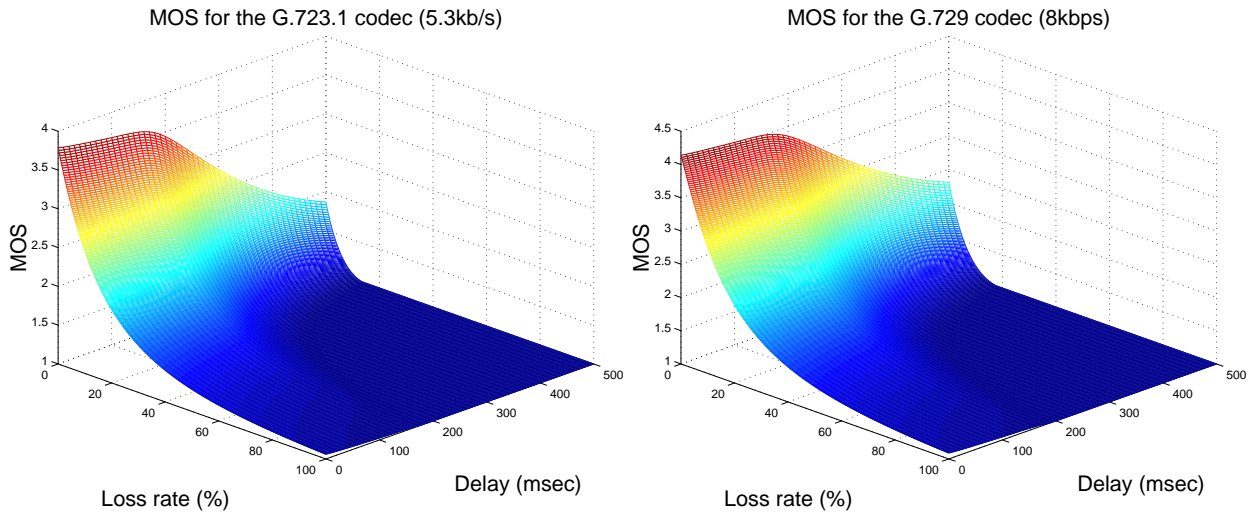


Figure 2.12: MOS of the G.723.1 and G.729 codec as a function of network delay and mean loss rate

voice quality than delay, quality degrades rapidly for loss rates higher than 5%. Note, however that using a different codec with different playout buffering and loss concealment schemes will yield a different relation between network loss, delay, and quality.

2.5 Conclusions

In this chapter, we have overviewed the basic mechanisms related to the voice media coding and transport over the Internet. We note that in VoIP systems, the media exchanged can be chosen entirely by end systems. As such, end systems can control the amount of compression based on network bandwidth. Furthermore, sending audio as packets makes it easy to suppress silence periods, further reducing bandwidth consumption and increasing efficiency of voice packet switching. Unfortunately, compression is at odds with enhanced voice quality services. Indeed, speech codecs has several important features, including sampling frequency, bit rate, complexity, delay and robustness to packet loss that affect the quality of voice delivered by VoIP systems. The bandwidth saving comes at the cost of lower quality and lower robustness to congested network environments.

We have also investigated the main impairments that cause VoIP quality degradation. In fact, recent voice codecs use PLC techniques and can tolerate some small loss ($< 5\%$) without severe quality degradation. However, voice traffic has unacceptable performance if long delays (> 150 ms) are incurred. The real-time transport of VoIP traffic performed by the RTP protocol and is augmented by a control protocol (RTCP) in order to offer insight on the performance and behavior of the media stream, such as delay, loss rate and jitter; however, RTP does not guarantee any quality of service for the transported voice media.

The ITU-T E-model, presented in this chapter, provides a framework that can be used to objectively assess the VoIP communication quality based on real-time network performance measure-

ments (i.e., delay and loss) and application-specific characteristics (i.e., audio codec). We notice that packet loss and delay affect the perceived quality of a VoIP communication in a non-trivial manner. Factors affecting overall packet loss and delay come from both the network and the application.

In the remainder of this manuscript we will focus on the case of large number of VoIP communications transmitted simultaneously over heterogeneous IP network environments. We investigate performance limitations in this case and we present our proposed mechanisms aiming to improve VoIP transmission quality and scalability.

Chapter 3

Towards a Congestion Control for VoIP Traffic

Summary

In this chapter, we investigate the performance limitations in the case of a large number of long distance VoIP calls originating from different sources and transported through a best-effort IP network. We focus on the potentially negative effects of the transmission of large number of small VoIP packets. These negative effects are two fold: on one hand protocol header overhead causes inefficient use of bandwidth. On the other hand, uncontrolled UDP bursts of short voice packets flowing into the IP network cause network congestion, which degrades the real-time VoIP transmission performance and creates a fairness problem because of TCP traffic being suppressed. We propose a new generic scheme called Voice-TFCC (Voice TCP-Friendly Congestion Control) that controls the packet and codec rate of VoIP flows. The main idea behind our contribution is to adjust the packet rate of VoIP connections by multiplexing many VoIP flows over a single stream and also to adapt voice codec rate based on TCP-friendly decision. Flow multiplexing reduces the traffic load (i.e., packet rate) on Internet routers and also reduces the overall header bandwidth used by VoIP. By this congestion control mechanism we let VoIP behave as TCP streams and the network transports hence homogeneous connections. A prototype version of the proposed protocol was implemented and tested over PlanetLab network [77]. Experimental results show that the proposed scheme for adjusting the packet and codec rate achieves efficient voice flow transmission and improves voice transmission quality while being fair to TCP.

3.1 Introduction

In best-effort packet switched networks, like the Internet, end-to-end congestion control is the key issue to guarantee fair sharing of the available bandwidth among competing flows [55]. The most popular instance of congestion control is that of TCP, where an adjustment of sending rate based on a sliding window concept is implemented. Congestion control has revealed its benefits also for the transport of multimedia traffic [61]. Current congestion control mechanisms are packet-based, they vary the packet/time rate according to network congestion state. The problem arises since delay-sensitive applications such as VoIP typically use UDP as transport protocol, not implementing any type of congestion control, in order to exploit the available bandwidth most efficiently. In fact, when these applications share network resources with other applications that use a congestion control algorithm (i.e., TCP flows), a poor utilization of the bandwidth usually results leading, in the worst case, to a starvation of the network resources. The said problem has stimulated new ideas from both protocol and network designers, which follow two approaches, not necessary exclusive of each other. One approach is to implement an intelligent queue management at Internet routers considered as the point of congestion. An important example of this approach is the Random Early Detection (RED) queue management [64] that address network congestion in a responsive rather than reactive manner. The second approach is to implement an end-to-end congestion control specifically conceived for real-time applications. The study of congestion control mechanisms is usually based on empirical considerations about the dynamics of window-based systems. Several techniques have been proposed to understand the in depth nature of congestion control [62][13]. They rely on mathematical models to catch the relevant aspects of a general congestion control problem, but these techniques are usually very complex to implement.

We address, in this part of our thesis, the need to design congestion control for the growing class of VoIP traffic. This need is motivated by the inefficient use of network bandwidth caused by protocol header overhead of voice packets and the fairness problem caused by the transmission of large number of small VoIP packets sharing network links with TCP traffic.

We propose a new VoIP congestion control scheme called Voice-TFCC (Voice TCP-Friendly Congestion Control), which tries to keep the transmission protocol overhead to a minimum while maintaining a TCP-friendly throughput. Voice-TFCC combines RTP voice flow multiplexing and the TCP-friendly congestion control mechanism. The approach developed in this contribution is to adapt the transport protocol and the way in which it interacts with the network in order to support voice flows competing with TCP traffic in the Internet environment. Voice-TFCC mechanism is applied on VoIP flows transmitted between two intermediate gateways. It adjusts the packet and codec rate in a TCP-friendly manner. Flow multiplexing is used to reduce the traffic load (i.e., packet rate) on Internet routers and to reduce the overall header bandwidth used by VoIP. Experimental and analytical results show that Voice-TFCC achieves efficient voice flow transmission and considerably improves VoIP transmission quality while being fair to TCP. To the best of our knowledge our proposal is the first scheme that incorporates packet and codec rate adaptation for RTP flows while maintaining TCP-friendliness.

The chapter is organized as follows. Section 3.2 begins with the problem statement. Following, in Section 3.3 we discuss related work. Section 3.4 presents the Voice-TFCC scheme. In Section 3.5, we analyze the dynamics of our proposal. Section 3.6 illustrates Voice-TFCC performance. Finally, Section 3.7 concludes the chapter.

3.2 Problem Statement

In order to reduce bandwidth usage, low-bit-rate voice codecs are used in IP telephony systems (Table 3.1). The commonly used are G.711 [17], G.729A [18], and G.723.1 [19].

Table 3.1: Low bit rates generated by three standard IP telephony speech codecs

Codec	G.723.1	G.729A	G.711
Bit rate (kbps)	5.3/6.3	8	64
Frame interval (ms)	30	10	20
IPv4/UDP/RTP header (bytes)	40	40	40
Payload (bytes)	20/24	20	160
IP bandwidth (kbps)	15.96/16.96	24	80
Overhead	66%/ 62.5%	66%	20%

When silence suppression scheme is employed, the codecs then operate in two states: a silent state at zero bit-rate and an active state at the compressed bit-rate. Regardless of the state, the frame period and frame size are still fixed [68].

After the digitization and compression operations, the voice frames are transmitted through the IP network. Since VoIP applications care more about time of delivery than reliability, they typically use the UDP transport protocol. UDP does not provide the underlying support for quality of service but it gives more flexibility in addressing application-specific requirements. Most of the current real-time applications over the Internet are based on UDP and the Real-time Transport Protocol (RTP/RTCP) [8][9] which is rather an application layer protocol. Although RTP does not contain any mechanism that guarantees the timely delivery of data, RTCP reports provide a regular summary statistics (loss rate, network delay and jitter) in order for the sender and the receiver to adapt their behavior to the current network congestion state (e.g, adapting sending rate at the sender side and adjusting playout buffers at the receiver so that random delays resulting from other network traffic can be compensated). RTP is currently widely used for multimedia communication and particularly for IP telephony in the Internet. Voice packets are encapsulated with IP/UDP/RTP headers (Figure 3.1). Therefore, for a packet duration of 20 ms it can be assumed that header information will add 16 kbps to the bandwidth requirement for voice over IP. For example, if an 8 kbps algorithm such as G.729A is used, the total bandwidth required to transmit each voice channel would be 24 kbps leading to an overhead of 66% (as shown in Table 3.1).

When IPv6 is used the header overhead will be larger due to IPv6 header of 40 bytes alone (60 bytes for total IPv6/UDP/RTP header). The header could also be larger due to IP options or IP

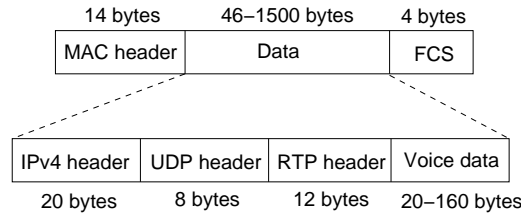


Figure 3.1: Voice packet encapsulation

tunneling mechanism that encapsulates IP packets with an additional IP header.

In Figure 3.2, we show the header overhead increase when the payload size of packets is decreased and also when large protocol header is used (e.g., IPv6 header).

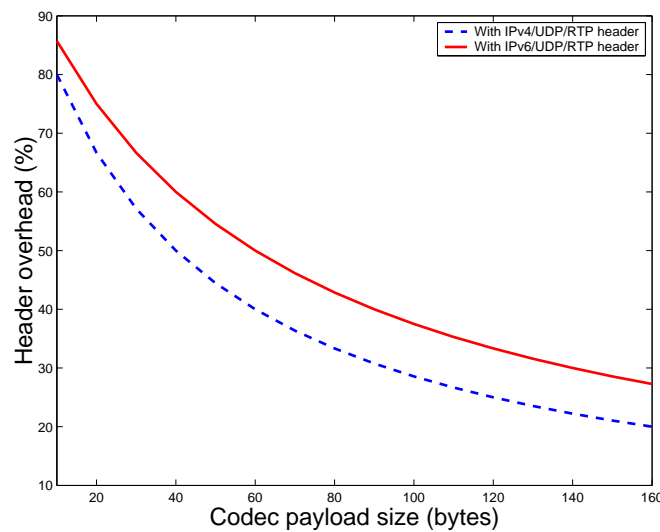


Figure 3.2: Protocol header overhead according to codec payload size

In fact, the voice codec defines the size of the sample but the total number of samples placed in a packet affects the rate of packets sent per second. This number is another factor in determining the bandwidth of a voice call.

A number of samples representing 30 ms is considered to be the maximum duration for the payload. This duration is a compromise between bandwidth requirements and quality. By changing the number of samples included in a packet this number, the amount of bandwidth a call uses can change definitely, but there is a trade-off.

On one hand, by increasing the number of samples, payload size will increase and the total number of packets sent per call will be reduced. The number of packet headers required for the call will be also reduced and the needed bandwidth will be decreased. In fact, smaller payloads demand higher bandwidth per channel band and decrease transmission efficiency, given that the minimum header length remains at 40 bytes (IPv4/UDP/RTP header).

On the other hand, when the payload size increases, the overall latency of each call also in-

creases since voice samples have to be buffered for a longer period of time. When payloads are increased the system will also be more susceptible to the loss of individual packets by the network.

Although, available bandwidth in current IP-based backbones is often sufficient enough for a limited number of VoIP communications, a large number of voice communications transmitted between two edge gateways of an IP backbone network will cause inefficient use of network resources, such as buffers and bandwidth due to packet header overhead and bursts of small voice packets.

3.2.1 Studied Architecture

In this work, we focus on the case of a large number of voice sources at an access network, sharing a common path and destined to different users in remote networks. IP backbone networks represent an important part of the end-to-end path for long distance VoIP calls, including calls that are serviced by a combination of a switched telephone network in the local area and the Internet for the long haul (Figure 3.3).

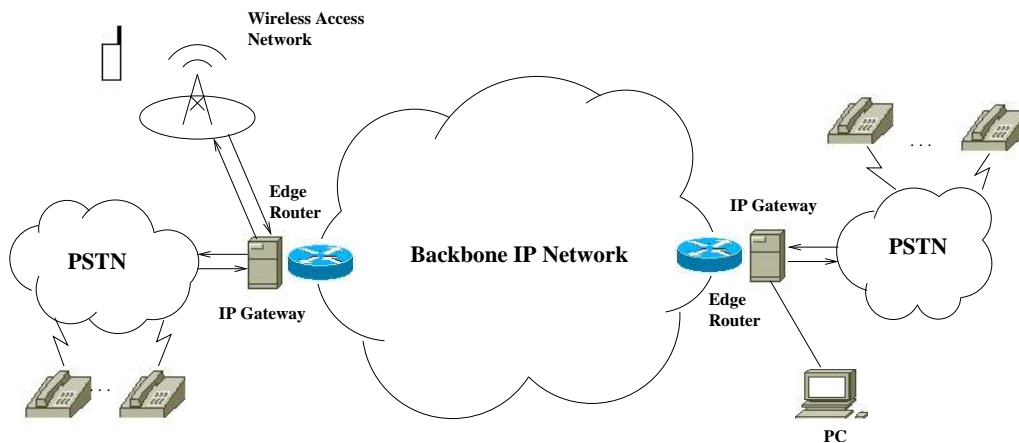


Figure 3.3: Heterogeneous mix of data and real-time services transmitted on the same IP infrastructure

In the considered architecture, we suppose the existence of a gateway between the communicating entities. In addition to phone-to-phone communication, PSTN/Internet gateways can be used to provide RTP supported phone-to-PC and PC-to-phone communications since RTP is integrated into the H.323 [33] and SIP [31] protocol stack. Call signaling protocols can be used in conjunction with data transport protocols to build a complete multimedia architecture. Typically, these architectures will include protocols such as RTP for transporting real-time data and providing QoS feedback, the Media Gateway Control Protocol (MEGACO) [57] for controlling gateways, and the Session Description Protocol (SDP) [32] for describing multimedia sessions.

3.2.2 Performance Limitations

As we discussed above, high-compression voice codecs used in IP telephony systems improve bandwidth efficiency enormously. However, the payloads are relatively small compared to the ad-

ditional overhead imposed by the network to transport the audio data between the sender and the receiver. This overhead results in a decrease of the usable band. For instance, a typical IP voice packet consists of a header of 40 bytes. Compared with the typical payload size of only 10 – 30 bytes for each audio frame, the header overhead is clearly very substantial. Current VoIP applications tackle this problem by embedding multiple audio frames into a single packet at the source to increase the ratio of payload to header size. This approach has the benefit of reducing the overall data rate of a call. But, since an audio frame is generated only after raw audio signals in a frame period are captured and encoded, packing an additional audio frame will add another frame period to the assembly delay. Together with the existing network delay, the resultant end-to-end delays may become unacceptable [68].

In local area networks where bandwidth is abundant, VoIP applications can send each audio frame in a separate RTP packet to minimize packetization delay. However, in case of Internet telephony gateways with multiple RTP streams, the bandwidth that the header occupies must be taken into consideration. Especially on backbone facilities where costs are high (e.g., some global cross-sections).

For example, carrying Voice over IP headers for the entire voice load of a large network with 300 million or more calls per day could consume on the order of 20 – 40 gigabits per second on the backbone network for headers alone [65].

From another point of view, UDP traffic is unresponsive to congestion and therefore can completely monopolize the available bandwidth. Thus, Internet load increases because of large numbers of short voice UDP packets, with 100 packets flowing every second in both directions for each call into the IP network, eventually resulting in large delay, jitter and packet loss [69]. Performance problems will be experienced, in this case, by all voice calls and also by other traffic (i.e., TCP traffic) sharing the best-effort IP network. We expect that this is also true for networks where differentiated scheduling algorithms, as DiffServ [7] are implemented, as the number of connections increases.

3.3 Discussion of Related Work

We can distinguish two main approaches that have been undertaken in the literature to tackle the header overhead problem:

- Reduce the header size (i.e., header compression)
- Encapsulate several packets in one header (i.e., multiplexing)

In this section, we discuss: the IP/UDP/RTP header compression standard, the multiplexing schemes of RTP voice flows and the TCP-friendly congestion control mechanism for unresponsive flows. We have coupled the two latter mechanisms to develop our multiplexing scheme.

3.3.1 IP/UDP/RTP Header Compression

VoIP packets are composed of one or more voice codec samples or frames encapsulated in 40 bytes of IP/UDP/RTP headers. 40 bytes is a relatively large amount of overhead for the typical 20 – byte VoIP payloads, particularly over low-speed links.

To solve this problem, header compression was introduced. An IP/UDP/RTP header compression scheme for low speed serial links was proposed by Casner and Jacobson in RFC 2508 [66]. Their scheme specifies compressed RTP (cRTP) which is designed to reduce the IP/UDP/RTP header size from 40 bytes to a minimum of 2 bytes for most packets in the case where no UDP checksums are being sent, or 4 bytes with checksums. The compression algorithm defined in this document draws heavily upon the design of TCP/IP header compression as described in RFC 1144. Two formats of cRTP are actually specified:

- Compressed RTP (CR) - used when the IP, UDP, and RTP headers remain consistent. All three headers are compressed in this case.
- Compressed UDP (CU) - used when there is a large change in the RTP timestamp or when the RTP payload type changes. The IP and UDP headers are compressed, but the RTP header is not.

Casner's algorithm takes advantages of two properties in RTP streams. First, most of the fields in the IP, UDP and RTP headers do not change over the lifetime of an RTP session. These constant-value fields can be represented by fewer bits with a session context during transmission. Second, RTP header fields like sequence number and timestamp are increased by a constant amount for successive packets in a stream. Hence, differential coding can be applied to compress these fields into few bits. However, cRTP operates on a link-by-link basis. These compression schemes are not widely used because of incompatible hardware.

Another header compression framework called ROHC (RObust Header Compression) was standardized in RFC 3095 [67]. ROHC specifies a highly robust and efficient header compression scheme for IP/UDP/RTP headers which is designed specifically for links with significant error rates and long round-trip times. RHOC aims at resolving the problem of the large header overhead related to IP over cellular links when used for interactive voice conversations.

Recently, a multiplexing scheme for RTP streams was proposed as a standard in RFC 4170 [73], it combines the IP/UDP/RTP compression with a Point-to-Point Protocol (PPP) multiplexing scheme [71] to form a method for end-to-end tunneling of multiplexed RTP packets over a network path for the purpose of reducing the bandwidth used when multiple RTP streams are carried over that path. Nevertheless, this proposal operates in the link layer and can only be used over PPP links¹.

¹Point to Point Protocol (PPP) is utilized to establish a connection between a PC and an ISP (Internet Service Provider). It is used on two points-connection only, usually modem to modem. PPP is specified in RFC 2153.

3.3.2 RTP Multiplexing Schemes

Multiplexing aims at the reduction of the overhead associated with Internet protocol layers and traffic load (i.e., number of packets) on routers. The basic assumption for the multiplexing is that, at any time, there is more than one user communicating with the same remote location. Various approaches for multiplexing VoIP streams between peer gateways have been proposed [68][70][74][75][76]. The basic idea is to multiplex voice calls between two gateways into a single packet, instead of using a separate connection and thus separate packets for each.

In [70], the authors propose to multiplex RTP streams sharing the same destination gateway into one UDP packet at a multiplexing interval period. This solution is simple and there is no additional header for multiplexing; it is effective for both header overhead and number of packet reduction. In typical case, 40% of the bandwidth is saved for 8 multiplexed G.723.1 encoded voice streams by header overhead reduction and number of voice packets also decreases 1 by 8.

The proposed scheme in [75] consists in merging several audio payloads coming from different users into a single packet. Only one IP/UDP/RTP header and a new RTP mini-header is added to the packet. The mini-header is required to delineate the multiplexed packets. The protocol adds 16 bits of overhead per multiplexed user. Although this scheme allows a big reduction of the header, it adds some protocol complexity and requires modifications of the RTP packet format defined in [8].

A light-weight data driven multiplexing approach is introduced in [76]. The basic idea is to replace the IP/UDP/RTP header of each packet with a mini-header at the edge router. The mini-header is a two-byte tag that replaces the original header at the ingress router, and will be used to reconstruct the original header at the egress router using a mapping table kept at each of the access routers. A control signaling protocol is also proposed to exchange simple control signals between peer entities.

The total number of voice-packets that are multiplexed should be obtained by considering trade-offs between delay and bandwidth efficiency improvements. Indeed, the main drawback of these multiplexing schemes is the added delay. Two types of delay are incurred:

- Multiplexing processing delay occurs in the gateway given that the generation of a multiplexed packet is triggered by the expiration of the multiplexing period or by the arrivals of enough voice packets to fit into the MTU. The multiplexing period is decided by the implementer of an IP gateway. If the chosen multiplexing period is small, the additional delay becomes small but the number of users in a multiplexed channel also becomes small [70]. It is possible to adjust the multiplexing period according to the number of existing connections so that the system can support all the voice calls with the smallest possible delay to attain the best-possible conversation quality.
- The multiplexed packet will also encounter additional packet transmission delay at Internet routers since most network limitations in today's Internet are in bytes per second (drop tail queues are in units of bytes per second). The router store-and-forward delay is proportional to the packet length and inversely proportional to the link speed, the larger multiplexed packets will introduce a longer transmission delay than normal voice packets.

With regard to packet loss, multiplexing is likely to reduce losses on the Internet. First, because improved efficiency results in a reduced overall bit rate for the voice stream and thus the necessary bandwidth. Secondly, many routers drop packets not because of link congestion and buffer overflow, but because of processing overload. A burst of small packets can overwhelm the processors on a typical router, causing losses.

We should note that RTP mixers described in Section 2.2.2 can not be used in the considered architecture where we deal with many unicast VoIP communications established between separate users. In fact, RTP mixers are used to reduce bandwidth usage in a multicast communication. The mixer resynchronizes incoming audio packets and mixes the reconstructed audio streams relative to different sources into a single stream and forwards the formed packet stream to a single recipient or to multiple recipients multicast. The RTP header includes a means for mixers to identify the sources that contributed to a mixed packet so that correct talker indication can be provided at the receivers. This is provided using the contributing source list (CSRC) that identifies all the contributing sources for the payload contained in this packet and consequently helps the participants to distinguish who happens to be speaking at any given time. However, de-multiplexing interleaved streams generated from the contributing sources remains unfeasible.

3.3.3 TCP-Friendly Rate Control for Unresponsive Flows

Unresponsive flows are flows that do not use end-to-end congestion control and, in particular, that do not reduce their load on the network when subjected to packet drops. This behavior can result in both unfairness and congestion collapse for the Internet.

TCP-friendly equation-based rate control for unresponsive flows inside best-effort networks was introduced [78] to ensure proper congestion avoidance for multimedia applications using unresponsive transport protocols, i.e., UDP and RTP, while coping with the real-time needs of these applications. That is, in contrast with the behavior of TCP, to smoothly find available bandwidth, increase the sending rate slowly in response to a decrease in the loss event rate and to do not halve the sending rate in response to a single loss.

The basic decision in designing equation-based congestion control is to choose the underlying control equation. In [79], a TCP-friendly flow is characterized by an upper bound of its arrival rate. This bound is given by the maximum overall sending rate for a TCP connection with a given packet drop rate, packet size, and round trip time. Given a packet drop rate of p , the maximum sending rate of a TCP connection, in absence of timeouts, for application packet size of S bytes with a fairly constant round-trip time, including queuing delays, of R seconds is given by:

$$T(\text{Bps}) \leq c * \frac{S}{RTT * \sqrt{p}} \quad (3.1)$$

with

$$c = 1.5 * \sqrt{\frac{2}{3}}$$

However, this simple model fails to accurately estimate TCP throughput under high loss rates. A more complex model proposed in [82] can be applied for a broader range of network conditions. The Equation 3.2 developed in this model, roughly describes TCP's sending rate as a function of the loss event rate, round-trip time and packet size.

The loss event rate p , between 0 and 1, is given by the number of packet loss events as a fraction of the number of packets transmitted. A loss event is one or more packets lost in an RTT, similarly to TCP which typically only performs one halving of the congestion window during any single RTT. In addition this equation takes into account TCP's window reduction events caused by timeouts, as this dominates TCP behavior at higher loss rates [83].

$$T(\text{Bps}) = \frac{S}{\text{RTT} \sqrt{\frac{2p}{3}} + \text{RTO} (3\sqrt{\frac{3p}{8}})p(1 + 32p^2)} \quad (3.2)$$

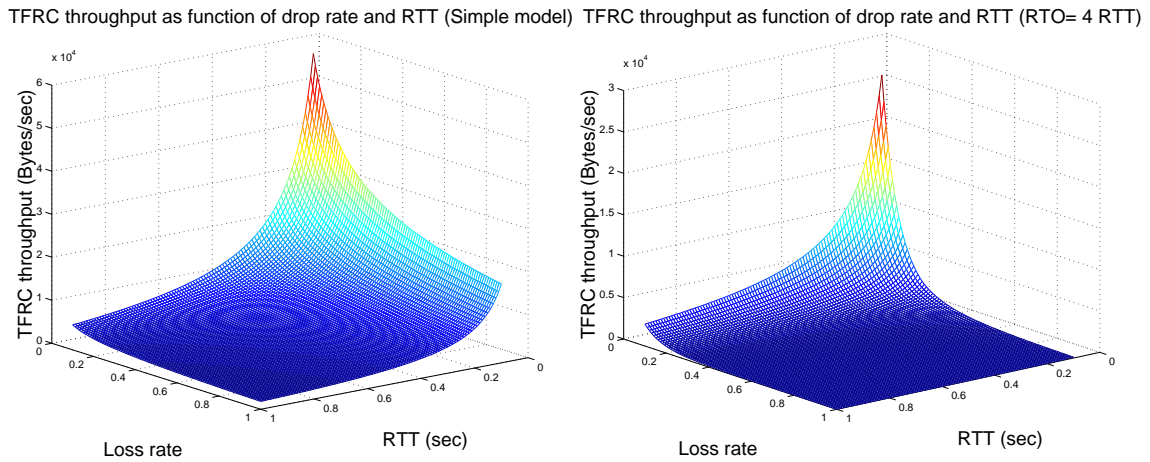


Figure 3.4: TFRC throughput as function of drop rate and RTT (payload size = 1460 bytes)

Figure 3.4 compares the TFRC throughput generated using TFRC Equations 3.1 and 3.2 respectively. The throughput model given by Equation 3.2 is clearly more conservative than the simple model of Equation 3.1, and particularly more sensitive to packet delay. The throughput given by the simple model for RTT values between 0 and 200 ms is considerably higher than that generated by the Equation 3.2 model.

Based on the TCP-friendly rate control mechanism, many schemes have been developed in order to provide appropriate congestion control for real-time applications in the Internet environment.

In the receiver-based multicast congestion control scheme introduced in [80], the receivers compute round-trip times, estimate the packet loss rate p , and use Equation 3.1 to compute the rate at which they should receive data.

In [81], the authors propose an end-to-end rate adaptation scheme suitable for unicast applications, that adjusts the transmission rate of multimedia applications to the congestion level of the network. Based on the estimation of the loss rate and the round-trip times obtained from the regular information of RTCP reports [8], the sender increases the transmission rate during network under-load periods and reduces this rate during congestion periods, while avoiding an aggressive adaptation behavior.

The joint rate, error, and delay control method for VoIP proposed in [12], includes a TCP-friendly module which controls the application data rate, while keeping the packet rate constant.

Currently, standardization efforts are in progress to design a TFRC mechanism which is appropriate for VoIP traffic. In a recent work [72], Floyd et al. proposed a VoIP variant of TFRC for applications that transmit small packets. The design goal of this variant is to achieve the same bandwidth in bytes/sec as a TCP flow using constant 1500 – byte data packets.

Floyd's scheme is motivated by the approach reported in RFC 3714 [63] which considers that it is acceptable to assume a network bandwidth limitation in bytes/sec rather than in packets/sec in considering the sending rate of telephony traffic.

This VoIP TFRC variant differs from the original TFRC scheme in the loss event estimation. The loss event rate is calculated by counting at most one loss event in loss intervals longer than two round-trip times, and by counting each packet lost or marked in shorter loss intervals, in contrast with the original TFRC scheme. In addition, instead of sending one large packet per round-trip time, the VoIP variant of TFRC could be sending N small packets (where N small packets equal one large 1500 – byte packet).

The goal of the VoIP variant of TFRC has been for the TCP flows and the VoIP TFRC flows to have rough fairness in the sending rate in bytes per second, in a scenario where each packet receives roughly the same probability of being dropped.

In a scenario where large packets are more likely to be dropped than small packets (i.e., byte-based queues), or where flows with a bursty sending rate are more likely to have packets dropped than are flows with a smooth sending rate, flows using the VoIP variant of TFRC could receive more bandwidth than competing TCP flows.

In order to avoid such effect, this TFRC variant was restricted to applications that send packets no more than once every 10 ms aiming to limit the sending rate in packet per second and to prevent a single flow from sending small packets arbitrarily frequently. In this scheme, the transmit rate X obtained from Equation 3.2 is also multiplied by a reduction factor according to Equation 3.3 in order to penalize VoIP applications that send small payload packets increasing header overhead.

$$X = X * \frac{S_{\text{true}}}{(S_{\text{true}} + H)} \quad (3.3)$$

Rate reduction factor

where S_{true} is the true average data packet size for the VoIP connection and H is the protocol header size. Figure 3.5 displays the reduction factor used by Floyd's scheme for an IPv4/UDP/RTP header of 40 bytes and an IPv6/UDP/RTP header of 60 bytes. Clearly, the lower the payload size is

the more sending rate will be reduced.

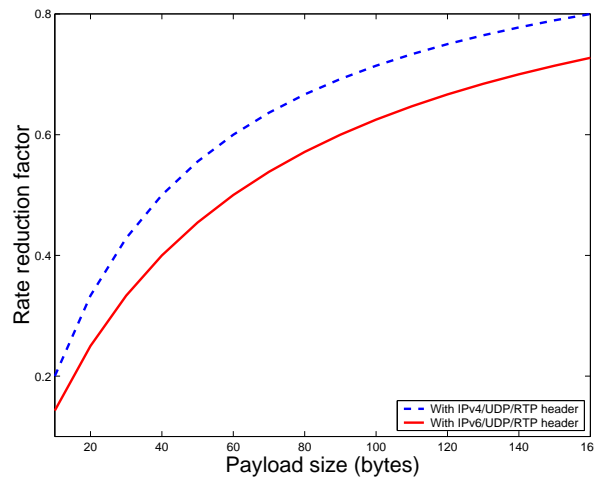


Figure 3.5: Rate reduction factor decreasing for small payloads

Figure 3.6 plots simulation results reported in [72] and performed using NS simulator. These 100 – second simulations show drop rates and sending rates of TCP and VoIP TFRC flows in a drop-tail packet-based queue environment. In a drop-tail queue in units of packets, each packet requires a single buffer, regardless of the packet size.

In these simulations, five TCP connections and five VoIP TFRC connections competing with web traffic over a 3 Mbps shared link. The VoIP TFRC application generates 200 – byte data packets every 10 ms giving a sending rate of 160 Kbps. The five TCP connections have round-trip times from 40 to 240 ms, and the five TFRC connections have the same set of round-trip times. Different levels of congestion are created by increasing the number of web sessions sharing the same bottleneck link. The average TCP and TFRC sending rates are averaged over the five flows .

In a packet-based bottleneck bandwidth, the decision to drop a packet is independent of the packet size. As shown in Figure 3.6, the VoIP TFRC flows see similar packet drop rates as the TCP flows, though the VoIP TFRC flows receives higher throughput than the TCP flows with packet drop rates of 25% or higher [72].

However, the fairness results can change significantly if the drop-tail queue at the bottleneck link is in units of bytes rather than packets.

For a queue in packets, the queue has a fixed number of buffers, and each buffer can hold exactly one packet, regardless of its size in bytes. For a queue in bytes, the queue has a fixed number of bytes, and an almost-full queue might have room for a small packet but not for a large one [72]. Hence, large packets are more likely to be dropped than are small ones.

In a drop-tail queue in bytes, as Figure 3.7 shows, the VoIP TFRC flow sees a much smaller drop rate than the TCP flow, and consequently receives a much larger sending rate.

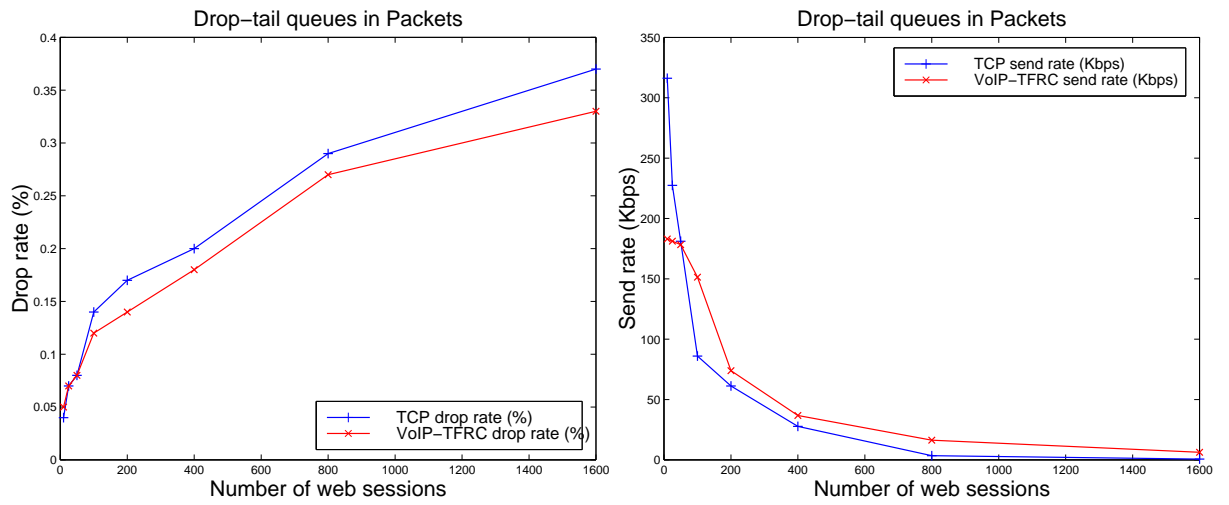


Figure 3.6: Drop rates and sending rates of TCP and VoIP TFRC flows in a drop-tail packet-based queue environment

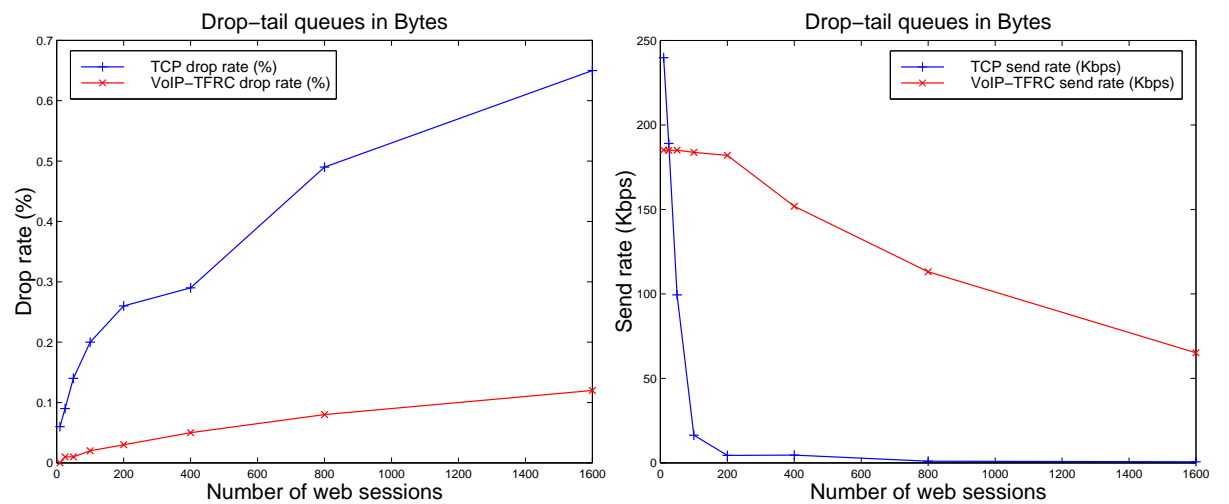


Figure 3.7: Drop rates and sending rates of TCP and VoIP TFRC flows in a drop-tail byte-based queue environment

These results lead to an interesting conclusion on how to support congestion control for VoIP flows that we exploited to design our VoIP congestion control scheme detailed in the Section 3.4 below.

3.4 Proposed VoIP Congestion Control Scheme

As stated in RFC 3714 [63], the ideal would be to have a transport protocol that is able to detect whether the bottleneck links along the network path are limited in bytes/sec or in packets/sec, and to respond appropriately, but such an ideal is hard to achieve. Thus, the deployment of congestion control for telephony traffic should not be delayed until such an ideal could be accomplished.

In this work, which is an improved version of our scheme proposed in [34], we design a new generic protocol named Voice-TFCC (Voice TCP-Friendly Congestion Control) that incorporates techniques for controlling both packet rate and codec rate of VoIP flows based on TCP-friendly congestion control mechanism. This protocol is applied to many VoIP flows that are transmitted between two VoIP gateways but can also be used by a single VoIP flow.

3.4.1 Voice-TFCC Scheme for RTP VoIP Flow Transmission

Voice-TFCC scheme is based on the idea of multiplexing an adjustable number of voice packets, this number is adapted to the current network congestion state. Since voice over IP applications use standard voice codecs which are unresponsive to congestion indication, Voice-TFCC adapts the control traffic (i.e., packet headers) of unresponsive voice flows to the network congestion state using TCP-friendly equation-based rate control mechanism. Voice-TFCC uses a simple multiplexing scheme described in [70], where header overhead is reduced through multiplexing several RTP streams destined to the same gateway into one UDP packet. Multiplexed packets are generated from different sources and occur at the same instant into the sending gateway (Figure 3.3).

The multiplex is formed by linking a series of RTP streams and an IP-UDP header (Figure 3.8).

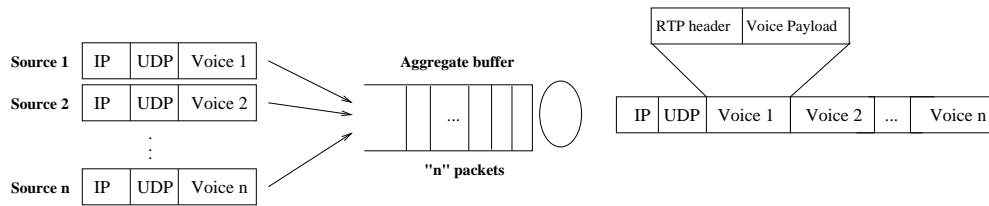


Figure 3.8: Packets from different RTP flows multiplexed into one UDP packet

Multiplexed packets are sent using one of the UDP ports designated for the multiplexed streams. At fixed time intervals, the receiver computes the loss rate observed during the previous interval. The sender, based on the receiver's feedback information, updates its sending rate by adjusting the number of packets to multiplex hence the packet rate (Figure 3.9). The number of RTP flows to

multiplex is limited by the maximum transfer unit packet size (MTU size = 1500 bytes considering protocol headers).

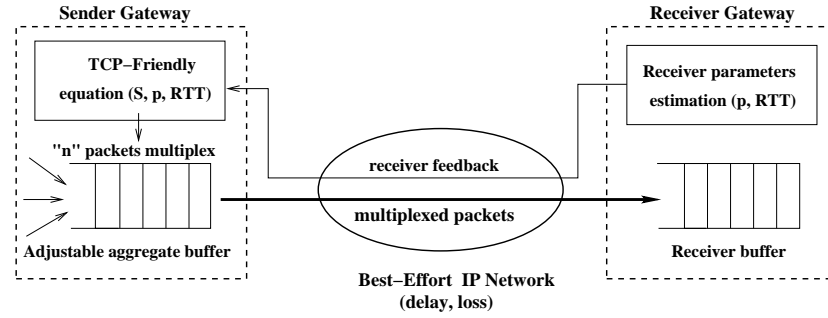


Figure 3.9: Adjusting the multiplexing buffer size in response to traffic fluctuations using the TCP-friendly congestion control mechanism

3.4.1.1 Voice-TFCC Sender Gateway Functionality

Basically, the sender estimates the values for the round-trip time RTT and the retransmit timeout value RTO after each transmission time period i (time period between the reception of RTCP feedback $i - 1$ and i). Since, we consider that the network delay is affected by the packet size, we recommend that the sender and receiver gateways use sequence numbers of a multiplexed RTP flows to estimate RTT values or to use RTCP SR and RR reports as explained in Section 2.2.3.1 while setting the packet size to the instantaneous packet size of the multiplexed RTP flow. The retransmit timeout value, RTO, is estimated from RTT. In practice the simple empirical heuristic of $RTO = 4 RTT$ works reasonably well to provide fairness with TCP [84]. The payload size of incoming voice frames at the sender gateway varies depending on the voice codec used by one source. Typical payload size is 20-160 bytes for each voice frame (Table 3.1). Our scheme incorporates a transcoding module at the Voice-TFCC sender gateway in order to handle flows using the same voice codec rate and to adapt this rate according to TCP-friendly decision (Figure 3.10).

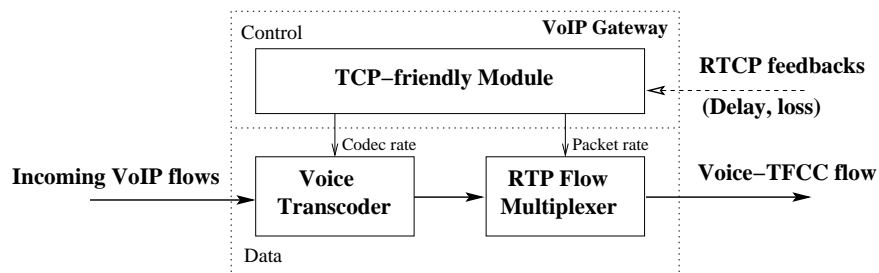


Figure 3.10: Voice-TFCC gateway architecture

Voice-TFCC scheme will then be applied to homogeneous voice flows generating the same codec throughput at a given transmission time period i . Voice-TFCC sender gateway adapts the number

of packets to multiplex and the codec throughput based on TCP throughput Equation 3.2 as the sending rate approximation, and according to Voice-TFCC algorithm (Algorithm 1) described in the Section 3.4.2 below.

Actually, Voice-TFCC sender adapts the throughput generated by protocol header in response to congestion while maintaining a steady voice data throughput in order to achieve requirements of voice over IP applications where a smooth sending rate is of importance. Voice-TFCC responds to changes of network congestion by adjusting the number of multiplexed packets, hence adjusting the packet rate and also the number of transmitted packet headers. In case of high load conditions or low bandwidth links, Voice-TFCC will reduce codec rate in addition to packet rate since multiplexing is limited by the MTU size. If TCP-friendliness constraint can not be achieved on incoming VoIP flows when Voice-TFCC is applied (i.e., by reducing packet rate and codec rate) then VoIP traffic load should be reduced. This can be done by restricting incoming VoIP flows at the sender gateway: for example by blocking voice calls or by routing VoIP flows through an intermediate gateway depending on network conditions on Internet paths. This intermediate gateway will forward VoIP packets from source gateway to destination gateway and will be chosen based on adaptive path selection technique that aims to find a routing path which is better than the one between source and destination gateways in terms of network conditions (i.e., network bandwidth, congestion state). This part of the scheme will be deeper studied in our future work.

3.4.1.2 Voice-TFCC Receiver Gateway Functionality

The receiver gateway periodically sends a feedback message reporting the loss event rate p to the sender. Each time the receiver sends a feedback message, it echoes back to the sender the sequence number of one RTP flow received in most recent multiplexed packet, and the time since that packet was received. The receiver keeps a packet history in order to detect loss of multiplexed packets. Packet loss is detected using RTP sequence numbers related to one multiplexed flow.

The estimated loss rate measures the loss event rate rather than the packet loss rate. A loss event can consist of several packets lost within a round-trip time, as it is discussed in [84].

We should note that in the studied case the sender gateway is transmitting at a high rate (many packets per RTT), but the receiver sends one feedback message per several multiplexed flows. This avoids bursts of control packets and improves the scalability of Voice-TFCC scheme. Voice-TFCC flow is de-multiplexed at the receiver gateway. Single voice flows could then be transcoded to fit network features and conditions between the gateway and final destinations. This mechanism is beyond the scope of this contribution.

3.4.2 Voice-TFCC Algorithm Description

The main idea behind our proposal is to decide the number of VoIP connections to be multiplexed over a single stream and the codec rate to be used based on TCP-friendly decision. The goal of this decision is to reduce the packet rate and hence to reduce the traffic load (i.e., number of VoIP packets) on Internet routers. Consequently, the header bandwidth used by VoIP flows will also

be reduced. When the maximum number of flows that can be multiplexed is reached, a second phase of the protocol is executed. This phase consists in transcoding VoIP flows in order to adapt the generated codec bit rate to the actual network bandwidth given by the TCP-friendly throughput estimation.

Let n denote the number of incoming VoIP flows at the sending gateway. Initially the number of packets to multiplex, m_0 , is set to 1 and the packet sending rate is determined by the total rate of voice flows coming at the sender gateway. When a feedback message is received, the sender changes the number of packets to multiplex based on the information obtained from the receiver gateway as explained in the following.

Assuming that at a transmission time period i the gateway is sending a flow of m_i multiplexed voice packets using codec_i and a total throughput rate of T_{TFRC}^i bytes/sec, then after receiving the feedback message RTCP_i from the receiver gateway the sender measures the round-trip time estimate, updates the retransmission timeout value. The loss rate obtained from the receiver gateway, p_i , and the measured round-trip time, RTT_i , are then fed into the throughput Equation 3.2, to give the new acceptable TFRC sending rate T_{TFRC}^{i+1} ($T_{\text{TFRC}}^{i+1} = T(S(m_i), \text{RTT}_i, p_i)$), where $S(m_i)$ is the size in bytes of a multiplexed packet formed by m_i RTP voice packets and one IP-UDP header, given by the Formula 3.4:

$$S(m_i) = h_{\text{ip}} + h_{\text{udp}} + m_i(h_{\text{rtp}} + \text{Payload}) \quad (3.4)$$

For a given number n_i of incoming VoIP flows to be transmitted between the sender and receiver gateway, the total throughput generated when m_i flows are multiplexed with one header is given by Equation 3.5:

$$B_{\text{mux}}^{\text{codec}}(i) = \underbrace{n_i T_{\text{codec}}^i}_{\text{codec throughput}} + \underbrace{\left\lceil \frac{n_i}{m_i} \right\rceil T_{\text{header}}}_{\text{header throughput}} \quad (3.5)$$

Where:

T_{codec}^i denotes the throughput in kbps, generated by the voice codec used (considering RTP header size and voice payload).

T_{header} denotes the throughput generated by IP-UDP header (i.e., 11.2 kbps)².

m_i denotes the number of RTP flows being multiplexed.

The basic idea behind the proposed mechanism is that the sender gateway should adjust the number of multiplexed RTP packets to have the throughput that matches the calculated rate T_{TFRC}^{i+1} in the next transmission time period ($i, i + 1$). We propose to use Equation 3.6 to determine the number of packets to multiplex according to the network congestion state:

²IPv4 header of 20 bytes and UDP header of 8 bytes, generated each 20 ms of voice frame period.

$$B_{\text{mux}}(i+1) = T_{\text{TFRC}}^{i+1} \quad (3.6)$$

where $B_{\text{mux}}(i+1) = n_{i+1} T_{\text{codec}}^i + \left\lceil \frac{n_{i+1}}{m_{i+1}} \right\rceil T_{\text{header}}$.

The network congestion state is reflected by the TFRC throughput estimation T_{TFRC}^{i+1} obtained after the transmission time period $(i-1, i)$. Basically, Voice-TFCC algorithm operates in two phases. In the Phase I, the codec rate is not changed between two successive time intervals. The number of flows to multiplex m_{i+1} is then determined according to the Equation 3.7:

$$m_{i+1} = \frac{n_{i+1} * T_{\text{header}}}{T_{\text{TFRC}}^{i+1} - n_{i+1} T_{\text{codec}}^i} \quad (3.7)$$

The sender will increase the number of multiplexed RTP packets, m_{i+1} , if there was a high traffic load during the previous time interval indicated by a calculated TFRC sending rate T_{TFRC}^{i+1} less than the previous sending rate T_{TFRC}^i . Otherwise, the sender will decrease the number of multiplexed packets during normal load periods.

If the resulting number of flows to be multiplexed is greater than the number of incoming flows ($m_{i+1} > n_i$) then all the incoming flows will be multiplexed ($m_{i+1} = n_i$) and the second phase of the Voice-TFCC algorithm is executed. The Phase II, consists in changing the codec rate in addition to packet rate. The codec rate to be used T_{codec}^{i+1} will be determined using the expression given by Equation 3.5 for ($m_{i+1} = n_i$) while considering codec rate as a variable:

$$B^{\text{codec}}(i+1) = T_{\text{TFRC}}^{i+1} \quad (3.8)$$

where $B^{\text{codec}}(i+1) = n_{i+1} T_{\text{codec}}^{i+1} + T_{\text{header}}$.

The new codec bit rate will be chosen among coding rates available at the sender gateway to best conform Equation 3.9:

$$T_{\text{codec}}^{i+1} = \frac{T_{\text{TFRC}}^{i+1} - T_{\text{header}}}{n_{i+1}} \quad (3.9)$$

The second phase of the algorithm is also executed if the resulting number of flows to be multiplexed is less than one flow ($m_{i+1} < 1$). This case indicates that the network bandwidth is sufficient for the transmission of n_i VoIP flows without the need for multiplexing and that the codec rate may be increased according to Equation 3.6 (m_{i+1} is set to 1) during Phase II of the algorithm.

The Phase II is executed in another case where the packet formed by multiplexing m_{i+1} flows using the codec throughput T_{codec}^i will exceed the MTU size. In that case, the m_{i+1} is set to the maximum possible number of flows to multiplex and codec bit rate is reduced using Equation 3.8.

Voice-TFCC scheme provides a generic TCP-friendly congestion control mechanism that can be used in the case of a single VoIP flow. The codec rate can be adapted using Equation 3.8. Figure 3.11 represents the flow chart of the Voice-TFCC scheme.

The pseudo-code of the basic Voice-TFCC algorithm is described below (Algorithm 1). Voice-TFCC framework parameters could be tuned according to the underlying network technology.

Algorithm 1 Pseudo-code of the Voice-TFCC algorithm

```

for a number  $n_i$  of incoming VoIP flows
  compute  $T_{TFRC}^{i+1}$  using RTCPi feedback
  if ( $n_i > 1$ ) {
    // Phase I: change packet rate by multiplexing adjustment
    Determine  $m_{i+1}$  from ( $B_{mux}(i+1) = T_{TFRC}^{i+1}$ )
    // Phase II: change codec bit rate in addition to packet rate
    if ( $m_{i+1} > n_i$ ) {
       $m_{i+1} = n_i$ 
      Determine  $T_{codec}^{i+1}$  from ( $B^{codec}(i+1) = T_{TFRC}^{i+1}$ )
    }else if ( $m_{i+1} < 1$ ){
      // Multiplexing is not needed
       $m_{i+1} = 1$ 
      Determine  $T_{codec}^{i+1}$  from ( $B^{codec}(i+1) = T_{TFRC}^{i+1}$ )
    }
  }
  else
    if ( $n_i == 1$ ) {
      // Adapt codec bit rate if there is only one VoIP flow
      Determine  $T_{codec}^{i+1}$  from ( $B^{codec}(i+1) = T_{TFRC}^{i+1}$ )
    }
  }

```

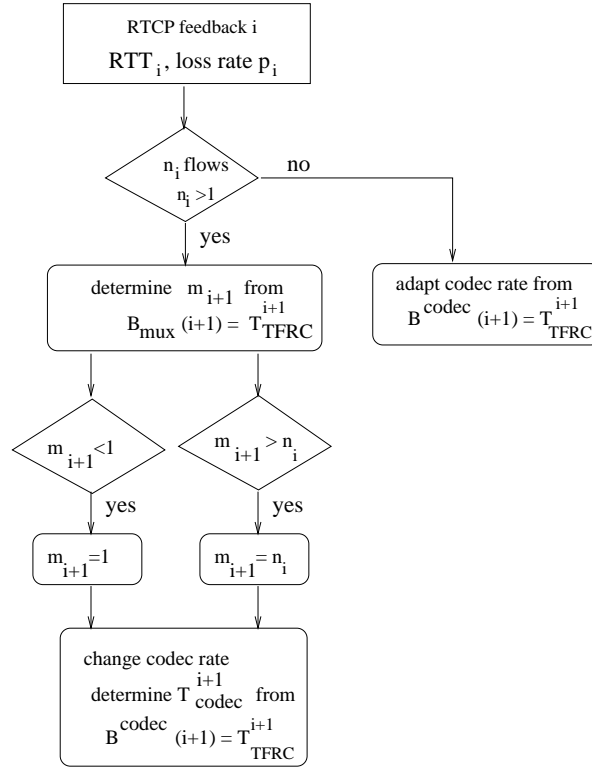


Figure 3.11: Flow chart of the Voice-TFCC scheme

For example, in the case of limited bandwidth on a wired Internet link, both coding and packet rate have to be adapted to have an optimal quality. However, on a link technology with large packet switching overhead such as in IEEE 802.11 wireless LANs, the packet rate should be lowered but the coding rate can be not reduced.

We should note that the proposed Voice-TFCC scheme can be used by RTP mixers and Translators (see Section 2.2.2) in the case of a multicast session (e.g., audio-conference) in order to adapt the packet rate and the transcoding rate of a multicast stream according to the network bandwidth related to a group of users.

3.5 Voice-TFCC Scheme Analysis

In this section, we discuss some issues related to the proposed Voice-TFCC scheme and we compare it to existing TCP-friendly congestion control and header reduction mechanisms.

3.5.1 Delay and Loss Rate of Voice-TFCC Packets

Since the multiplexed Voice-TFCC packet is formed by RTP frames originating from different sources and flowing at a given time at the sender gateway, packet delay will be increased just by a small multiplexing interval (< 20 ms) required for synchronization purposes between the different

VoIP sources. Actually, Voice-TFCC sends out a multiplexed packet every T_m (ms), which is equal to or shorter than the VoIP inter-packet interval (typically 20 ms). Larger values of T_m can improve bandwidth efficiency since more packets can be multiplexed, but the delay incurred will also be larger. For example, if $T_m = 10$ ms, every two multiplexed packet contains one voice packet from each VoIP stream. The maximum introduced multiplexing time for one voice packet is 10 ms. If $T_m = 20$ ms, every multiplexed packet contains one voice packet from each VoIP stream, and the maximum introduced multiplexing time is 20 ms. Adjusting the multiplexing time T_m , is another factor that can be used to control the tradeoff between bandwidth efficiency and delay. In this work, we will not study the dynamics related to this factor. We consider that this time is constant.

Although, packet loss rate may increase for large packet sizes especially in byte-based network environments, robustness against packet losses will not be affected given that for a given VoIP source, RTP packet (i) and RTP packet ($i + 1$) are transmitted in separate multiplexed IP packets.

3.5.2 Discussion of the Variable Size of Voice-TFCC Packets

Originally, TCP-friendly rate control mechanism was designed for applications that use fixed packet size, and vary their sending rate in packets per second in response to congestion. TCP-friendly rate control mechanism should not be used for applications that vary their packet size instead of their packet rate in response to congestion [83]. Varying the packet size during the time interval between two estimations of the sending rate distorts packet-based measurement of the loss event. Voice-TFCC adapts its sending rate by adjusting the number of multiplexed packets, consequently the packet size is varied. However, Voice-TFCC varies the packet size only after the estimation of the sending rate using the TCP-friendly throughput equation and keeps this size fixed until the next feedback message. Therefore, Voice-TFCC sending rate estimation is quite accurate.

3.5.3 Comparison of Voice-TFCC Scheme with Related Schemes

The TCP-friendly rate control schemes proposed in [79][80][81] incorporate congestion control mechanism at the application level of one user, i.e., they instruct the applications at the end systems to adapt the bandwidth share they are utilizing to the network congestion state. In contrast with these schemes, our Voice-TFCC scheme is applied at the transport level on a flow formed by multiplexed RTP voice flows originating from different sources between two edge IP gateways.

The common point between TFRC-based schemes that have been proposed in the literature is that they are applicable to single flows that are not implementing congestion control. Our contribution introduces a TCP-friendly congestion control scheme that is applied to many VoIP flows. It reduces protocol header overhead by controlling the packet rate (packets/sec) and reacts also to network congestion by adapting codec rate (bytes/sec).

Applications using the TFRC variant proposed in [72] could have a fixed packet size or could vary their packet size in response to congestion. We note however, that in this work, the authors does not specify any mechanism to be used by VoIP applications to adapt their throughput in response to the network congestion state in order to comply with the throughput given by the TFRC

equation.

The IP/UDP/RTP header compression scheme proposed in [66] reduces header size from 40 bytes to a minimum of 2 bytes in the best case (one condition is that UDP checksum from the source is disabled). The IP/UDP/RTP header compression method also relies on the link layer for exchanging control messages in order to preserve a loss-less compression: the IP header checksum is elided assuming that the link layer is providing good error detection (e.g., PPP's CRC). The total length field (in the IP header) is eliminated since it is considered as redundant with the length provided by the link layer.

While this method offers a full restoration of the IP/UDP/RTP header, it is link-by-link based as opposed to our scheme that combines application and transport layer mechanisms. Our scheme provides a simple header reduction method (concatenating voice packets from different RTP streams into a single UDP packet) in conjunction with an adaptive voice coding mechanism. Voice-TFCC scheme is not supported by the link layer, it is applied at the transport layer in which packets may traverse several links. Thus, Voice-TFCC is more suitable for latency sensitive voice traffic.

The innovation of our scheme, in comparison to the multiplexing schemes proposed in [68][70][74][75][76], consists in the adaptation of the number of multiplexed packets and also the codec rate (bytes/sec). This adaptability is controlled by the TCP-friendly rate control mechanism for unresponsive flows. By varying the number of multiplexed packets using Algorithm 1, Voice-TFCC sender adapts the throughput generated by protocol header in response to congestion while maintaining a steady voice data throughput. The number of multiplexed packets represent a compromise between the additional multiplexing delay and the number of users in a multiplexed channel (as discussed in Section 3.3.2).

By using TCP-friendly rate control, Voice-TFCC responds to changes of network congestion by adjusting the number of multiplexed packets and the codec rate. Hence, the smallest possible delay will be obtained. Moreover, being TCP-friendly reduces the network congestion and consequently reduces loss of voice packets. This will help to attain the best-possible conversation quality.

3.6 Evaluation of the Voice-TFCC Scheme

In this section, we illustrate through analytical and experimentation results the performance of our proposal.

3.6.1 Saving Bandwidth by Voice-TFCC Scheme

In a bandwidth limited network environment, the resource utilization will depend on the packet size. Without multiplexing, the bandwidth required for the transmission of n RTP voice packets is given by:

$$B_n = n * (h_{ip} + h_{udp} + h_{rtp} + \text{Payload}) \quad (3.10)$$

With Voice-TFCC scheme, the bandwidth required for the transmission of the same amount of voice data by multiplexing m RTP voice packets into one UDP packet (assuming that n is a multiple of

m) is given by:

$$B_{n,m} = \frac{n}{m} * (h_{ip} + h_{udp} + m * (h_{rtmp} + \text{Payload})) \quad (3.11)$$

The bandwidth saved by multiplexing m RTP voice packets can then be calculated from Equation 3.12:

$$\begin{aligned} \beta_m &= \frac{B_n - B_{n,m}}{B_n} \\ &= \frac{(1 - \frac{1}{m}) * (h_{ip} + h_{udp})}{h_{ip} + h_{udp} + h_{rtmp} + \text{Payload}} \end{aligned} \quad (3.12)$$

Figure 3.12 plots the percentage of bandwidth saving vs. the number of multiplexed packets for a typical payload sizes of 20 bytes and 160 bytes.

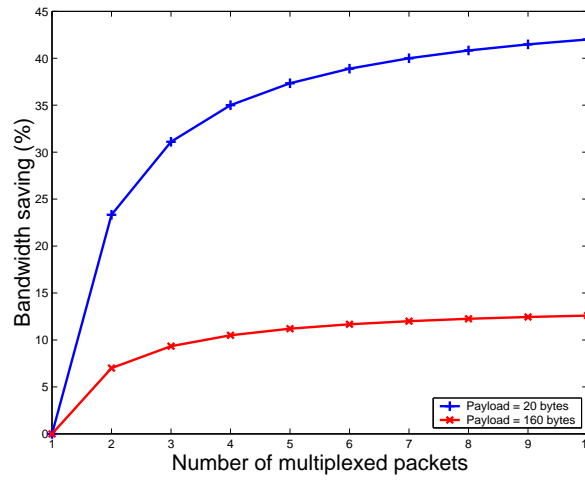


Figure 3.12: Bandwidth saving as the number of multiplexed packets increases

We notice that the bandwidth saved increases quite significantly when the number of multiplexed packets is varied from 1 to 10 packets. With 10 multiplexed packets having a payload of 20 bytes, a significant bandwidth saving of 42% is achieved. Note that the multiplexed-packet length is bounded by the MTU.

In Figure 3.13, we show the effect of payload size on the bandwidth saving obtained by multiplexing. It is important to notice that multiplexing smaller packets allows more bandwidth saving, for a given number of multiplexed packets, in comparison to the bandwidth required for the transmission of non-multiplexed packets. In the second phase of the Voice-TFCC algorithm, the codec rate is changed but the number of multiplexed packets is kept the same, thus the bandwidth gain is obtained from payload size reduction and not header overhead reduction.

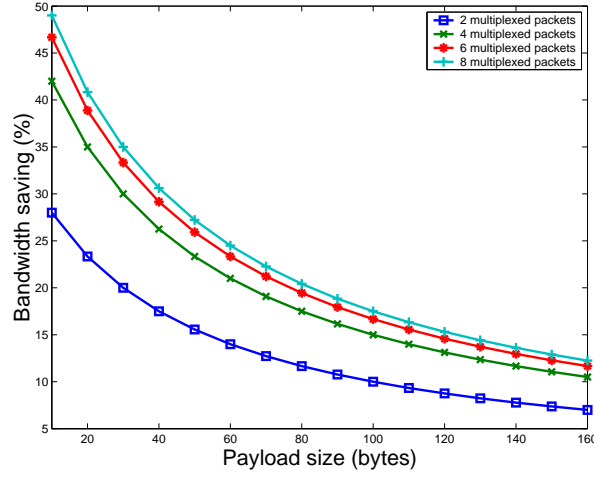


Figure 3.13: Bandwidth saving as the size of multiplexed packets increases

3.6.2 Experimental Results and Discussion

We implemented a prototype version of the proposed Voice-TFCC protocol and tested it with experiments over PlanetLab [77] testbed network. In this section, we present a representative set of experimental results performed using PlanetLab hosts listed in Table 3.2. PlanetLab hosts

Table 3.2: PlanetLab hosts used for experimental study

Host Id	Hostname	Site location	Bandwidth limit
host1(it)	planetlab1.polito.it	Torino (Italy)	1 Mbps
host2(usa)	planetlab1.nycm.internet2.planet-lab.org	New York (USA)	10 Mbps

impose outgoing network bandwidth limit, ranging from 500 Kbps to 10 Mbps, in order to limit the effect of application programs on other network users. Since the number of UDP flows that can be generated is limited by the machine capacity, we have chosen to run experiments using host1(it), having an outgoing network bandwidth limit of 1 Mbps, in order to investigate the effect of network bottleneck bandwidth on the performance of Voice-TFCC scheme. In practice, some Planetlab hosts were not usable due to certain ports being blocked, thus preventing UDP packets being sent to some sites.

The prototype implementation of VoIP traffic generation is based on UDP sockets. For the experiments, we have used an adaptive system that can switch between five bit-rates, corresponding to widely used voice coding standards as shown in Table 3.1 below. Packet voice streams are generated using constant bit rate (CBR) flows (see Annex A for VoIP traffic modeling). The number of sources generating VoIP traffic at the sender host (used to emulate the sender gateway) is increased through the decrease of the inter-arrival time of packets. For one VoIP flow we considered a packet inter-arrival time of 20 ms that corresponds to the frame size of typical voice codecs (Table 3.3). We assumed continuous voice packets arrivals that are enough to fit into multiplexed packets.

Table 3.3: Features of main standard VoIP codecs

Voice codec	Compression method	Bit rate (kbits/s)	Frame size (ms)	Payload size (Bytes)	Equipment Impairment I_e
G.711	PCM	64	20	160	0
G.726	ADPCM	24/32/40	20	60/80/100	25/7/2
G.729A	CS-ACELP	8	20	20	11

Under high load conditions, a high bit rate codec (i.e., G.711) is replaceable by a low bit rate codec (i.e., G.729A) without any significant quality sacrifice as shown in Figure 3.14. The quality is slightly less for I_e value of 11 (relative to G.729A codec) than the quality obtained for I_e value of 0 (relative to G.711 codec).

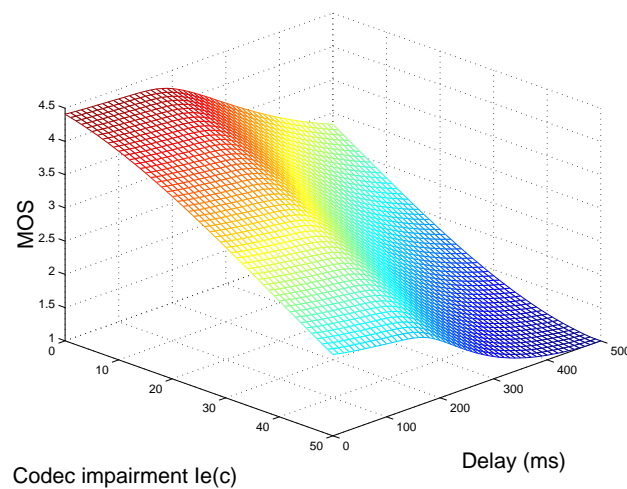


Figure 3.14: MOS as function of network delay and codec Impairment factor

Network parameter estimation (i.e., network delay and packet loss rate) is performed after the reception of feedback reports sent from destination host to the sender host every 5 seconds. This time interval affects the reactivity of the proposed scheme to network congestion state variations and is chosen to be compliant with RFC 3158 [60] recommendations for RTP testing strategies.

We used Equation 3.2 for TCP-friendly rate estimation and we applied Voice-TFCC algorithm on VoIP flows to determine the number of packets to multiplex and the codec rate to be used.

To measure VoIP quality, we applied the ITU-T E-Model described in Section 2.4. The R factor given by the E-Model is then translated to MOS quality measure using Equation 2.8. MOS value of 1 refers to the worst quality and the value of 5 represents the perfect call quality.

In the first experiment, 10 VoIP flows are transmitted from host1(it) to host2(usa) located at different Internet domains. Figure 3.15 plots traceroute RTT measurements for each network hop between these two hosts. These RTT values are less than the maximum permitted delay bounds for real-time audio transmission (i.e., 150 ms one-way end-to-end delay). However, we should note

that the network and application level RTT vary from each other considerably, signifying that the network-level RTT given by network measurement tools may not necessarily match the application level RTT due mainly to differences in routing paths and packet sizes.

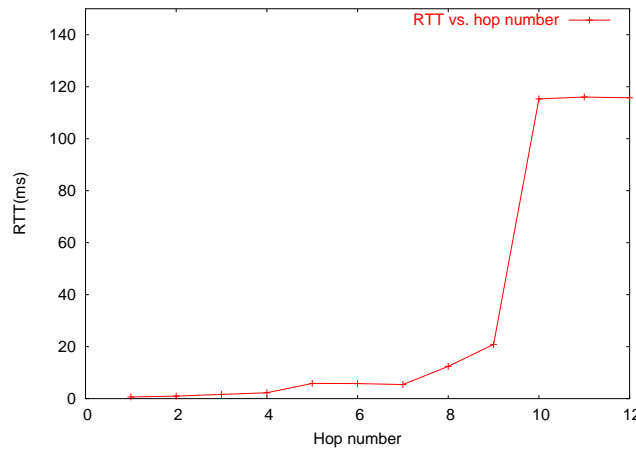


Figure 3.15: Traceroute RTT measurement for network hops between source and destination hosts

Initially, packets of 160-byte payload were generated (which represents the throughput generated by voice packets using G.711 codec). We have measured network parameters (i.e., delay, loss, jitter) of VoIP packets from packet sequence number and timestamp information contained in log files recorded at the sender and the receiver hosts (PlanetLab hosts have synchronized clocks). The jitter is calculated for each packet according to formula 2.4 reported in RFC 3550 [8]. Jitter is logically correlated to delay variations.

For the first experiment, Figures 3.16-3.18 compare the delay, jitter, and MOS values obtained using VoIP flow standard transmission with those obtained using Voice-TFCC mechanism. Figure 3.19 illustrates the rate difference between the Voice-TFCC total rate and the rate given by the TCP-friendly rate estimation.

In the second experiment, the same scenario as the first experiment was used with 5 VoIP flows transmitted from host1(it) to host2(usa). Figures 3.20-3.23 illustrate the same measurement entities as in the first experiment (i.e., delay, jitter, MOS and Voice-TFCC /TCP-friendly rate difference). Table 3.4 summarizes obtained results.

From these results, we made the following observations. Initially, before starting the execution of Voice-TFCC algorithm, VoIP flows experienced high delay values (more than 500 ms). This delay spike can significantly affect playout quality from an interactive viewpoint, it maps to a low quality MOS rating (< 2). After the reception of the first receiver feedback report, observed delay values were less than 200 ms during the remainder part of the transmission and MOS score was maintained at the value of 4 (Figure 3.18 (b) and Figure 3.22 (b)) which corresponds to a good VoIP quality. However, when Voice-TFCC is not used, experienced delay values (Figure 3.16 (a) and Figure 3.20 (a)) were well above the ITU-T recommended values of 150 ms resulting in a much increased mean delay value (1500 ms). These very high delays severely impairs voice quality reflected by MOS

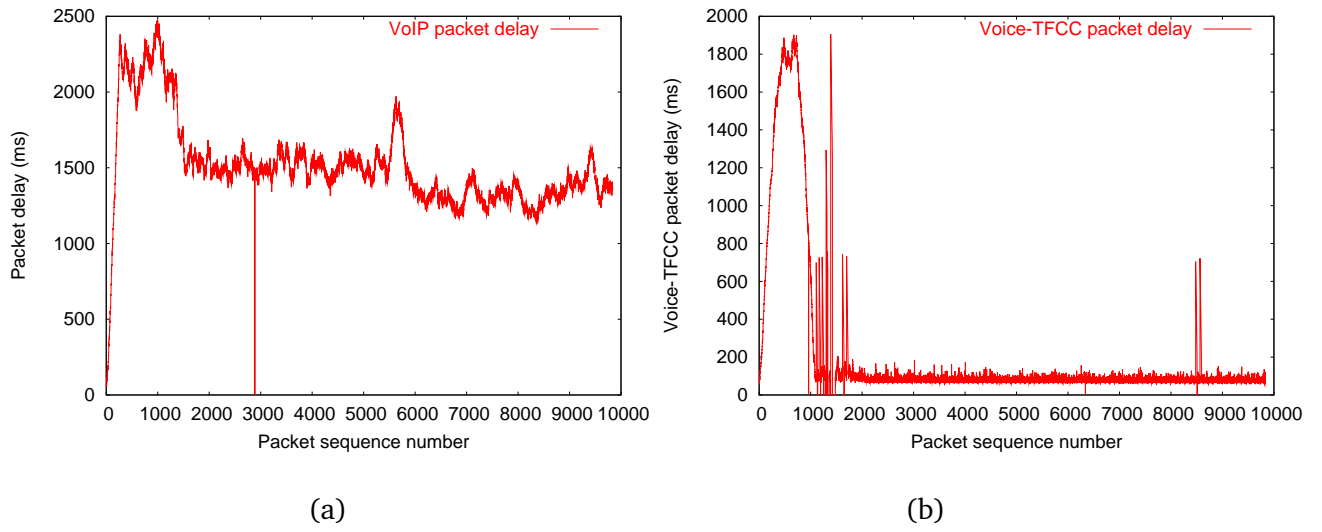


Figure 3.16: Packet delay of (a) VoIP flows vs. packet delay of (b) Voice-TFCC flow

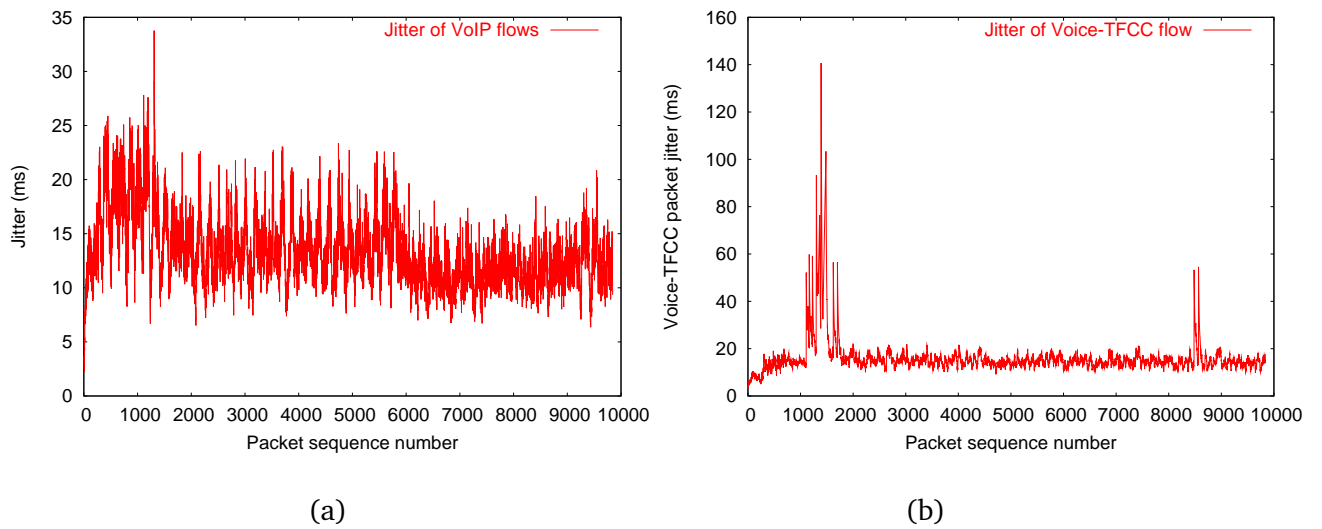


Figure 3.17: Jitter of (a) VoIP flows vs. jitter of (b) Voice-TFCC flow

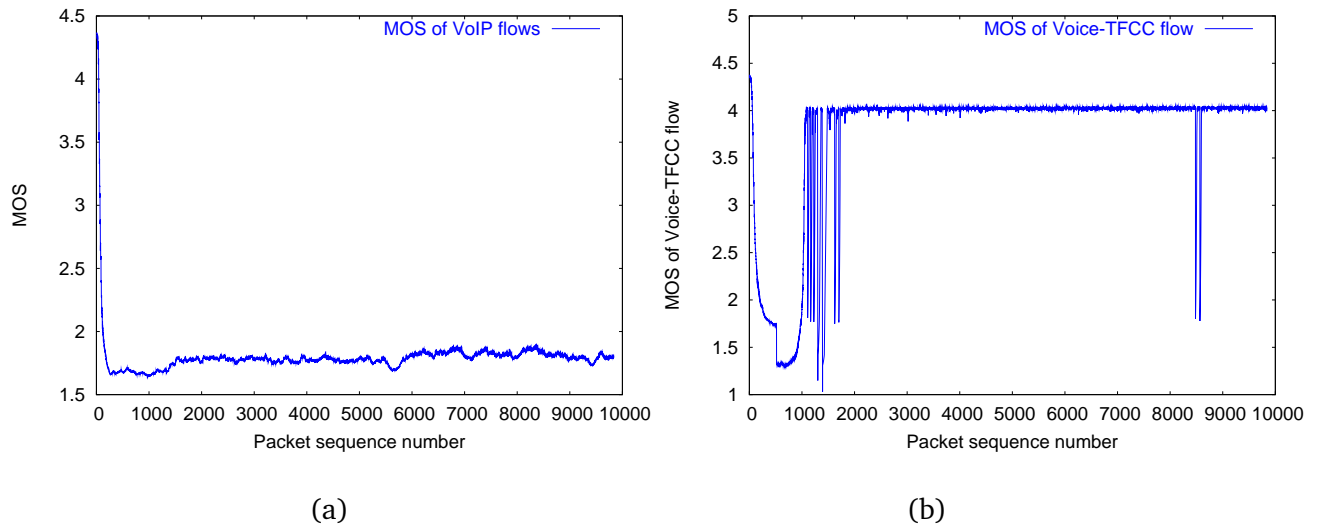


Figure 3.18: MOS of (a) VoIP flows vs. MOS of (b) Voice-TFCC flow

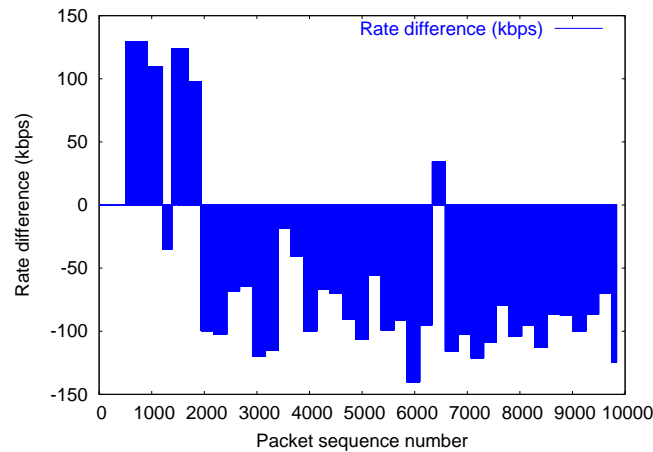


Figure 3.19: Rate difference between actual VoIP transmission rate and TCP-friendly rate estimation

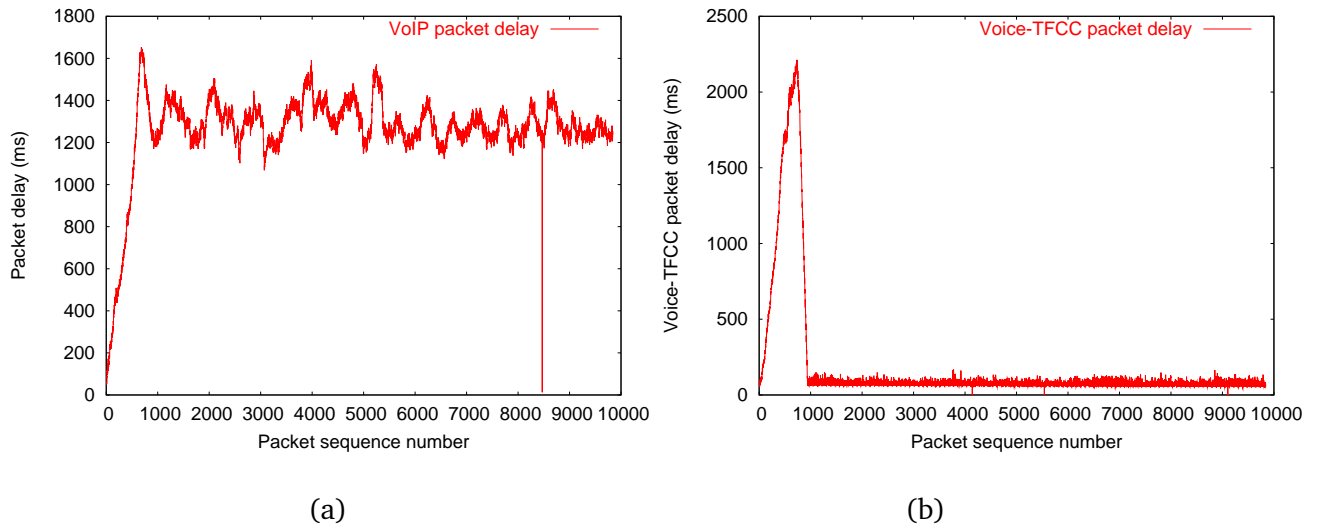


Figure 3.20: Packet delay of (a) VoIP flows vs. packet delay of (b) Voice-TFCC flow

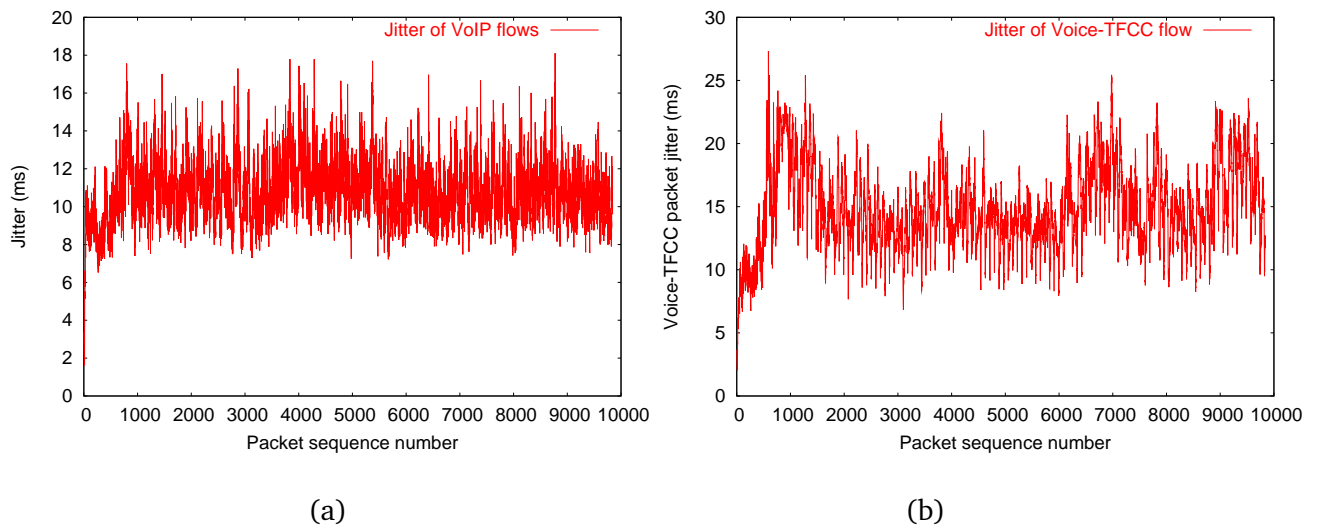


Figure 3.21: Jitter of (a) VoIP flows vs. jitter of (b) Voice-TFCC flow

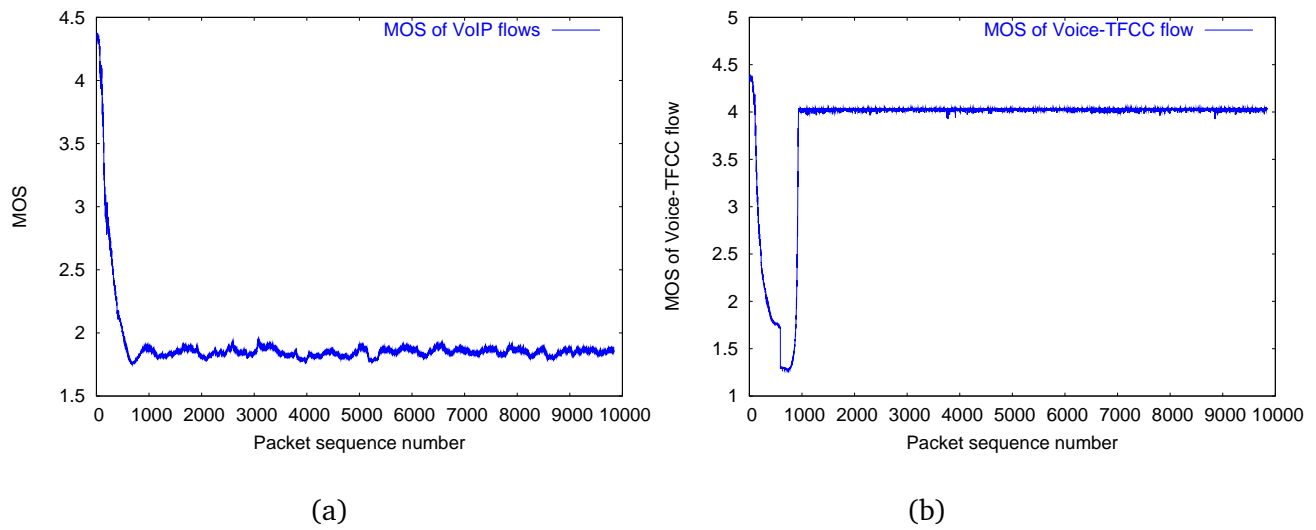


Figure 3.22: MOS of (a) VoIP flows vs. MOS of (b) Voice-TFCC flow

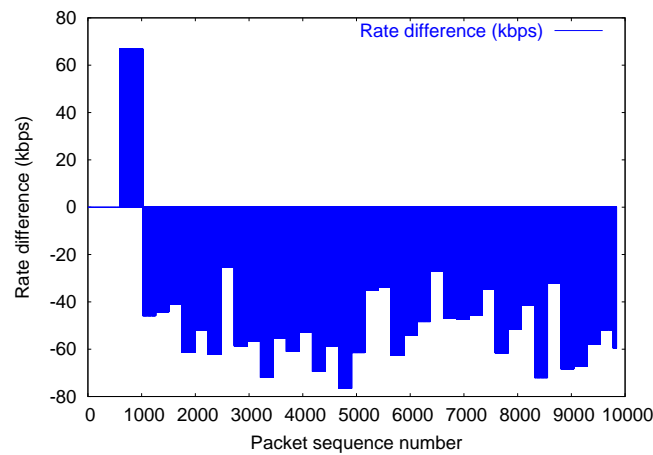


Figure 3.23: Rate difference between actual Voice-TFCC transmission rate and TCP-friendly rate estimation

Table 3.4: Results summary: with and without Voice-TFCC mechanism

Experiment	Number of flows	Mean delay (ms)	Mean loss (%)	Mean jitter (ms)	Mean MOS
VoIP flows	10	1501.91	0.1	13.21	1.85
Voice-TFCC flow	10	226.64	1.32	15.31	3.74
VoIP flows	5	1249.85	0.1	10.73	1.91
Voice-TFCC flow	5	183	0.3	14.85	3.85

Experiment	Number of flows	Mean number of multiplexed RTP packets	Mean payload size (bytes)	Mean packet size considering IP-UDP header (bytes)
VoIP flows	10	1	160	200
Voice-TFCC flow	10	9.52	27.42	340.11
VoIP flows	5	1	160	200
Voice-TFCC flow	5	4.76	28.36	188.71

rating less than 2 (Figure 3.18 (a) and Figure 3.22 (a)).

Another observation is that during these two experiments, the mean codec payload size and the mean number of multiplexed RTP packets (Table 3.4) indicate that the used codec was adapted to the lowest codec rate (corresponding to 20-byte frames of the G.729A codec) and that all incoming flows were multiplexed during the major part of the overall transmission time (10 flows in the first experiment and 5 flows in the second). Voice-TFCC dynamically decreases the header traffic load and the number of packets at Internet routers according to the network congestion state, therefore it decreases congestion and queuing delays.

Figures 3.17 and 3.21 show the jitter for VoIP flows and Voice-TFCC flow. When Voice-TFCC mechanism is used, the jitter is slightly higher than the case of standard VoIP flows transmission (mean values shown are in Table 3.4). The mean jitter value is 15.31 ms in the first experiment and 14.85 ms in the second when Voice-TFCC is used. These jitter values do not impair VoIP quality. Adaptive buffering at VoIP receiver can be used to overcome jitter variations.

We should note that the delay and jitter increase observed at the beginning of Voice-TFCC transmission can be avoided by sending probe packets before VoIP packets in order to assess the available network bandwidth and to estimate the TCP-friendly sending rate. This probing time period can be hidden for the user by inserting short waiting time (i.e., 5 seconds) before starting a VoIP communication.

Figures 3.19 and 3.23 illustrate the rate difference between the actual Voice-TFCC sending rate and the estimated TCP-friendly rate. Negative values indicate a Voice-TFCC sending rate that is lower than the allowed TCP-friendly rate. This is the desirable behavior for VoIP flows in order to compete fairly with TCP traffic.

The difficulty in Voice-TFCC adaptation is in the suitable determination of packet rate and codec rate that should match sending rate estimation given by TCP-friendly formula. Fine-grained Voice-

TFCC rate can be achieved by using a large spectrum of codec bit rates for transcoding or also by applying different codec bit rates for incoming VoIP flows in order to match more exactly the TCP-friendly sending rate.

In the third experiment, the same previous scenario was used with 5 VoIP flows transmitted from host1(it) to host2(usa). In this experiment, we have disabled the Phase II of the Voice-TFCC algorithm, in order to evaluate the effect of packet multiplexing without codec rate adaptation. The codec payload size was maintained constant (160 bytes) during the overall duration of the experiment. Figures 3.24-3.28 illustrate the same measurement entities as in the previous experiments (i.e., delay, jitter, MOS and Voice-TFCC /TCP-friendly rate difference). Table 3.5 summarizes obtained results.

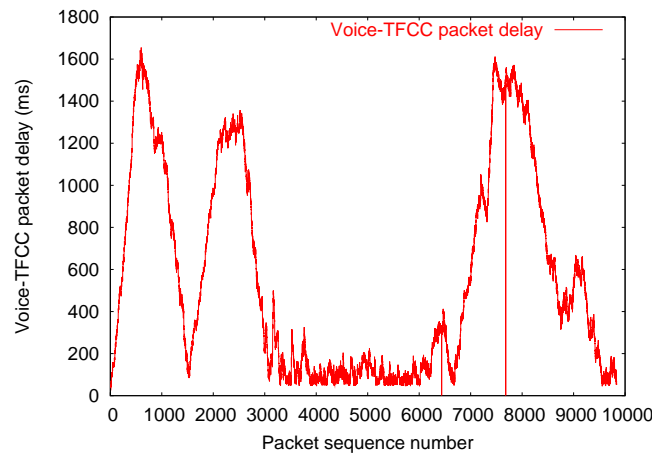


Figure 3.24: Packet delay of Voice-TFCC flow

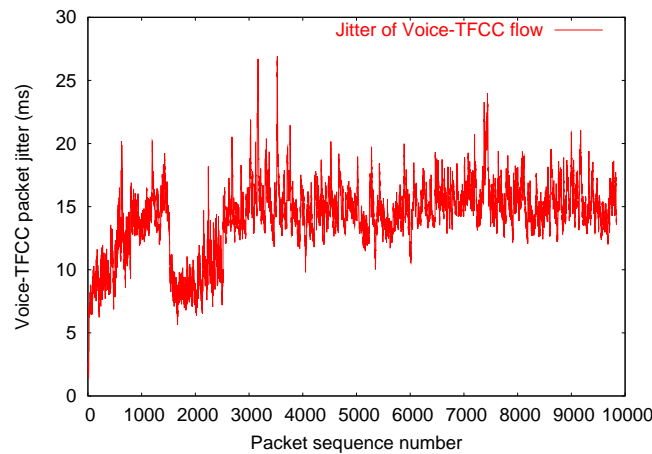


Figure 3.25: Jitter of Voice-TFCC flow

In the third experiment, high delay values (higher than 200 ms) occurred sporadically throughout the testing, resulting in a high mean delay value (560 ms). Consequently, a poor VoIP transmis-

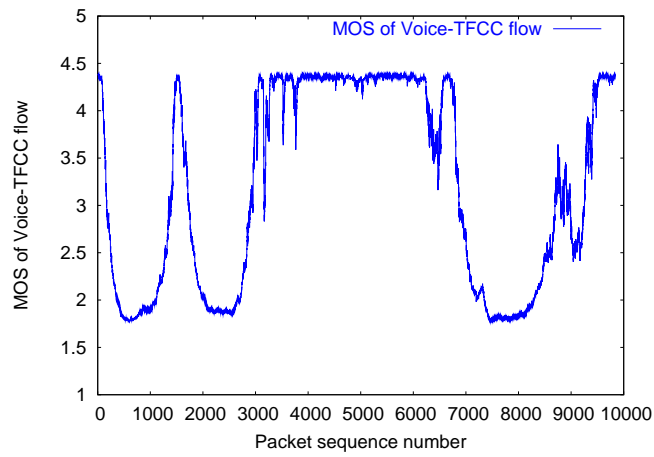


Figure 3.26: MOS of Voice-TFCC flow

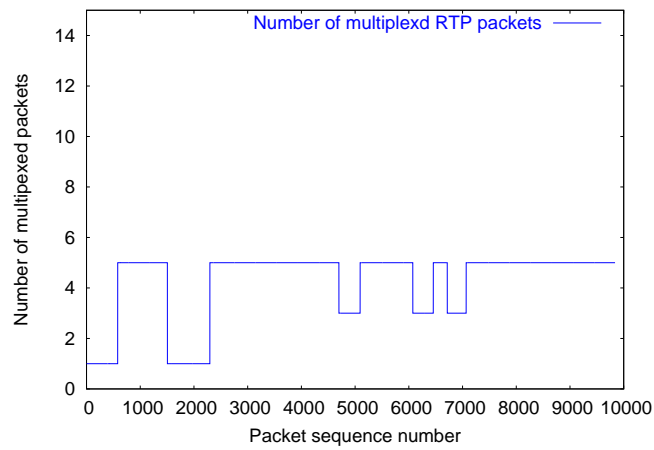


Figure 3.27: Number of multiplexed VoIP flows

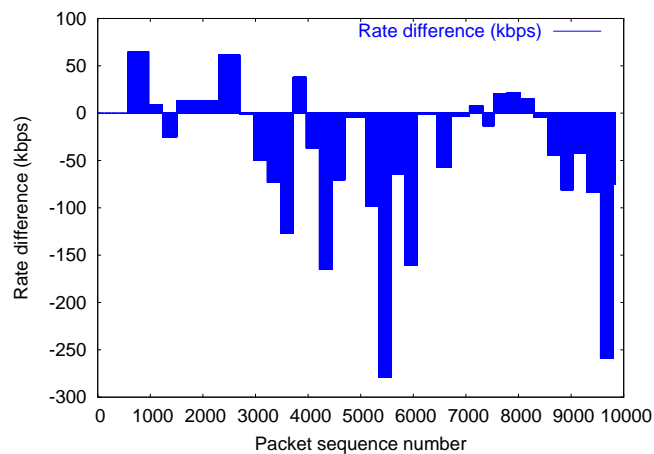


Figure 3.28: Rate difference between actual Voice-TFCC transmission rate and TCP-friendly rate estimation

Table 3.5: Results summary: Voice-TFCC mechanism without codec rate adaptation

Experiment	Number of flows	Mean delay (ms)	Mean loss (%)	Mean jitter (ms)	Mean MOS
Voice-TFCC flow	5	560	0.002	14.11	3.21

Experiment	Number of flows	Mean number of multiplexed RTP packets	Payload size (bytes)	Packet size considering IP-UDP header (bytes)
Voice-TFCC flow	5	4.21	160	200

sion quality was obtained (MOS value of 2). These high delays correspond to time periods when the number of multiplexed packets was reduced (Figure 3.27) in order to match the TCP-friendly sending rate estimation.

An interesting observation is that although the codec payload was fixed to 160 bytes, Figure 3.28 shows that during the major part of transmission time intervals Voice-TFCC transmission rate was lower than the TCP-friendly rate estimation. Moreover, the difference between TCP-friendly rate estimation and the Voice-TFCC rate is higher (more than 200 kbps) than that of the second experiment (60 kbps), as shown in Figure 3.23. In the second experiment the same scenario was considered but the codec payload was adapted according to the Phase II of Voice-TFCC algorithm. The mean payload size of the second experiment was 28.36 bytes. This result can be explained by the fact that the Equation 3.2 used for the TCP-friendly rate estimation is proportional to:

$$\frac{S}{RTT\sqrt{p}}$$

where S is the packet size and hence allows more sending rate when bigger packets are transmitted. However, in bandwidth-limited network environments, this behavior will be penalized by packet delay increase due to queuing delays occurring especially in case of network congestion. The delay increase will result in a reduction of the TCP-friendly rate estimation and hence in a reduction of the number of VoIP flows that can be accepted for the transmission between VoIP gateways. For that reason, codec rate adaptation is important. It allows the transmission of large number of VoIP flows between the VoIP gateways with a slight quality degradation due to the equipment impairment related to low bit rate codecs.

3.7 Conclusions

In this chapter we addressed the need to design congestion control for the growing class of VoIP traffic. We designed a novel scheme called Voice-TFCC (Voice TCP-Friendly Congestion Control) for joint control of coding and packet rate of VoIP flows. The packet rate is dynamically adjusted through multiplexing, the codec bit rate is adapted using different audio codecs. Thus, in both

bandwidth (bytes/sec) and packet rate (packets/sec) limited bottleneck environments, Voice-TFCC reduces network utilization. Moreover, packing more RTP frames in each IP packet reduces the number of packets sent and hence the overhead introduced by IP/UDP headers. Voice-TFCC scheme uses a TCP-friendly rate control mechanism in order to ensure fairness with current Internet traffic (i.e., TCP traffic).

The performance of our proposal was illustrated through a combination of analysis and real experimentation over PlanetLab network. Performance evaluation results show that Voice-TFCC achieves performance goals of voice flows (reduced delay and delay jitter) as well as efficient network utilization, and fairness with TCP traffic. Our scheme is scalable because no changes are needed at core routers and minimal control messages are used: one feedback message per several multiplexed voice flows. In addition, no modifications of the RTP packet format are required and no router support is necessary then it can be easily implemented and deployed in today's Internet. It is important to note that the proposed Voice-TFCC scheme is also interoperable with QoS mechanisms. For example, multiplexed packets can be marked with Expedited Forwarding (EF) bits for service differentiation. This will help to further decrease queuing delays at routers.

Chapter 4

Capacity Evaluation of VoIP in IEEE 802.11e WLAN Environments

Summary

Due to its convenience, mobility, and high-speed access, WLAN represents an important future trend for Internet access. At the same time VoIP market is taking off quickly given the advantages of VoIP technology. Thanks to the convergence of these two trends, VoIP over WLAN (VoWLAN) is becoming an important Internet application. Consequently, WLANs will need to support a large number of concurrent VoIP communications. For this motivation, we study VoIP capacity in WLAN environment. We propose an analytical model for VoIP capacity evaluation under HCF (HCCA and EDCA) mode of the IEEE 802.11e WLAN. We illustrate performance results relative to typical codec rates of G.711 PCM (64 kbit/s), G.729 (8 kbit/s) and G.723.1 (6.3 kbit/s). G.729 and G.723.1 allow greater capacity than G.711 which is constrained by throughput. We observe that the superframe (SF) size is a trade-off between the WLAN capacity and the voice packet delay. Modeling results also show that RTS/CTS mechanism reduces the VoIP capacity especially in the case of small voice frames.

4.1 Introduction

The motivation behind supporting VoIP over IEEE 802.11 WLAN environments comes from the spreading use of VoIP telephony by LAN users and the remarkable deployment of IEEE 802.11b [87] based wireless LANs (referred to as Wi-Fi). The main reasons for WLANs deployment are low equipment costs, the simplicity of setup and the high data rates. While 802.11b devices operate in the 2.4 GHz bands and provide data rates of 11 Mb/s, newer technologies such as IEEE 802.11a [88] promise higher data rates (up to 54 Mb/s) in the 5 GHz bands. IEEE 802.11g based WLANs [89] also support rates of up to 54 Mb/s, however, they operate in the 2.4 GHz band and are backwards compatible with 802.11b [100]. With the ongoing migration to wireless connectivity, VoIP users will represent a source of considerable VoIP traffic in WLANs. Interesting questions to ask are then: How many voice communications can be supported in a 802.11 WLAN ? What are the factors that dictate the 802.11 WLAN capacity ? and how to increase this capacity ?

Related research work (i.e., [92][95][97][100]) shows that a typical IEEE 802.11 WLAN with 11 Mbps bandwidth could only support a very limited VoIP connections in DCF/PCF mode. In [97], authors show that under IEEE 802.11b DCF mode the capacity of G.711 VoIP using CBR model and a 20 ms packetization interval is 12 calls. In this work, we study the VoIP capacity in IEEE 802.11e WLAN and to investigate increasing this capacity by reducing VoIP codec rate while maintaining an overall good quality. We introduce an analytical VoIP capacity model in order to assess the VoIP capacity for the upgrade version of IEEE 802.11e MAC standard. In our study, we analyze the capacity in both HCCA and EDCA modes. We show the effect of application level parameters (codec rate) and MAC parameters, i.e., superframe (SF) and contention period (CP) durations on the VoWLAN capacity. We also show the benefit of recent technologies such as IEEE 802.11a with high physical data rates (up to 54 Mb/s), that will allow a high VoIP capacity. This chapter is organized as follows: we first present the reference model for the IEEE 802.11 standard in Section 4.2. In Section 4.3, we analyze the IEEE 802.11e QoS enhancements. We study related work on VoWLAN capacity evaluation in Section 4.4. The studied network system is described in Section 4.5. In Section 4.6, we discuss protocol layers and headers overhead issues. In Section 4.7, we describe the proposed model for VoIP capacity evaluation over IEEE 802.11e networks and we analyze modeling results for different WLAN parameters. Finally, Section 4.8 concludes the chapter.

4.2 IEEE 802.11 MAC Standard

The IEEE 802.11-1999 standard [85] specifies two channel access mechanisms: A mandatory contention-based distributed coordination function (DCF) and an optional polling-based point coordination function (PCF). PCF is centralized and requires the presence of a base station that acts as an Access Point (AP). If PCF is supported, both PCF and DCF coexist and the time is divided into superframes. A superframe starts with a beacon frame. The AP generates beacon frames at regular beacon frame intervals; this time interval is announced in every beacon frame. Thus, every station knows when the next beacon frame will arrive.

Each superframe consists of a contention period (CP) where DCF is used and a contention free period (CFP) where PCF is used. During the CFP, the AP sends poll frames to high priority stations. Whenever a station is polled, it has the right to start transmission over the medium. To ensure that no station interrupts this mode of operation, the interframe space (IFS) between PCF data frames is shorter than the usual DCF interframe space (DIFS). This space is called a PCF interframe space (PIFS). To prevent starvation of low priority flows, the contention period must always be long enough for at least the transmission of one maximum frame length. The AP at any point during CFP period can suspend the PCF mode and return to DCF mode by sending a CF-end frame.

The DCF protocol is based on CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Each station (STA) maintains a Contention Window (CW) that is used to determine the number of slot times a station has to wait before starting transmission in case where the channel is busy. A backoff counter which is measured in slot times (see Table 4.1) and chosen randomly in the interval $[0, CW]$, begins to be decremented by one only after the medium has been free for a DIFS period. DIFS is equal to $SIFS + 2 \times \text{SlotTime}$. If the backoff counter expires and the medium is still free the station begins to transmit. The backoff counter stops when the channel is busy. In case of a collision the station randomly picks a new backoff period from its CW. CW is an integer whose range is determined by the physical layer characteristics: CW_{\min} and CW_{\max} as shown in Table 4.1 summarized from [88][87][89][100].

Table 4.1: Parameter values of IEEE 802.11a/b/g physical layer standards

MAC Parameters	802.11a	802.11b (DS) Direct Sequence	802.11b (FH) Frequency Hopping	802.11g-only	802.11g 802.11b-compatible
SIFS (μsec)	16	10	28	10	10
DIFS (μsec)	34	50	128	28	50
Slot Time (μsec)	9	20	50	9	20
CW_{\min} (Slot Time)	15	31	15	15	15
CW_{\max} (Slot Time)	1023	1023	1023	1023	1023
Frequency (GHz)	5	2.4	2.4	2.4	2.4
Supported data rates (Mbps)	6, 9, 12, 18, 24, 38, 48, 54	1, 2, 5.5, 11	1, 2, 5.5, 11	6, 9, 11, 18, 24, 38, 48, 54	1, 2, 5.5, 6, 9, 11, 12, 18, 24, 38, 48, 54

CW is doubled after each unsuccessful transmission, up to the maximum value which is determined by $CW_{\max} + 1$. When the backoff counter reaches zero, the source transmits the data packet. The ACK is transmitted by the receiver immediately after a period time called SIFS (Short Inter Frame Space) time which is less than DIFS. When a data packet is transmitted, all other stations hearing this transmission adjust their network allocation vector (NAV), which is used for virtual carrier sense (CS) at the MAC layer (Figure 4.1). The NAV maintains a prediction of future traffic on the medium based on the duration information that is announced in Data frames (or RTS/CTS frames as explained in the following) prior to the actual exchange of data. In addition, whenever a node detects an erroneous frame, the node defers its transmission by a fixed duration indicated by

EIFS (Extended Inter Frame Space) time. This time is equal to $SIFS + ACK_{time} + DIFS$ time. Due to collisions and the binary backoff mechanism, there are no transmit guarantees with DCF.

DCF includes a basic access method and an optional method with request-to-send (RTS) and clear-to-send (CTS) exchanged. If the optional access method is used, an RTS frame should be transmitted by the source and the destination should accept the data transmission by sending a CTS frame prior to the transmission of the actual data packet. STAs in the sender's range that hear the RTS packet should update their NAVs and defer their transmissions for the duration specified by the RTS. Nodes that overhear the CTS packet update their NAVs and refrain from transmitting. This manner, the transmission of packet data and its corresponding ACK can proceed without interference from other nodes (hidden node problem).

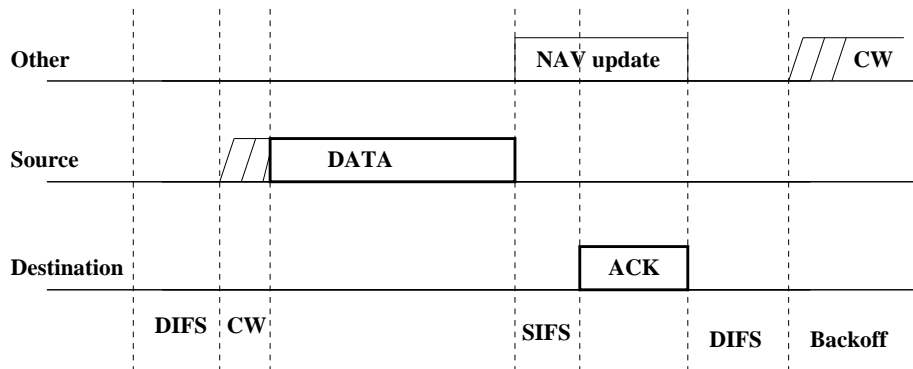


Figure 4.1: IEEE 802.11 DCF medium access scheme

With PCF, the period after each beacon transmission is divided into two sections, the Contention Free Period (CFP) followed by the Contention Period (CP), which together constitute a superframe. The point coordinator (PC), generally assumed to be co-located at the Access Point (AP), has guaranteed access to the medium in the beginning of the CFP. During the CFP, the PC lets stations have priority access to the medium by polling them in a round-robin fashion. Any station that requests to be added to the polling sequence is included during the next polling interval.

4.3 Analysis of IEEE 802.11e QoS Enhancements

IEEE 802.11 MAC scheme based on DCF/PCF mechanism provides a best-effort service which is not sufficient for achieving a reasonable quality in scenarios with high background load. This shortcoming in the 802.11 standard was addressed by the IEEE 802.11 Task Group E which has proposed a number of enhancements of the standard. The IEEE 802.11e standard [86] defines a super-set of features specified in the original standard and introduces the Hybrid Coordination Function (HCF). HCF uses both a contention-based channel access method, called the Enhanced Distributed Channel Access (EDCA) and a contention-free channel access method, called HCF Controlled Channel Access (HCCA) which have been derived from their earlier versions EDCF (Enhanced Distributed Channel Function) and HCF (Hybrid Coordination Function).

With the EDCA, QoS is supported by using four access categories (ACs), each corresponding to an individual prioritized output queue. With the HCCA, a hybrid coordinator (HC) allocates transmission opportunities (TXOPs) to wireless STAs by polling, so as to allow them contention-free transfers of data, based on QoS policies. An HC can generate an alternation of CFP and CP.

4.3.1 HCF Controlled Channel Access (HCCA)

Under HCF the basic unit of allocation of the right to transmit onto the wireless medium is the transmission opportunity (TXOP). Each TXOP is defined by a starting time and a defined maximum length. The TXOP may be obtained by a QoS Station (QSTA) winning an instance of EDCA contention during the CP, or by a non-AP QSTA receiving a CF-Poll during the CP or CFP. In an infrastructure network, the AP will gain access to the medium with a higher priority than other QSTAs.

An HC generates an alternation of CFP and CP, the sum of the two periods forms the superframe (SF). In addition, contrary to DCF, QSTAs can be polled during the CP in periods called Controlled Access Periods (CAPs) as shown in Figure 4.2. During a beacon interval, a QAP is allowed to start several CAPs using HCCA at any time after detecting the channel being idle for PIFS time interval. The duration of TXOPs allocated to each QSTA is determined by the HC scheduler according to the requested QoS parameters. It is also possible to change the SF length since beacons carry a parameter indicating the SF length.

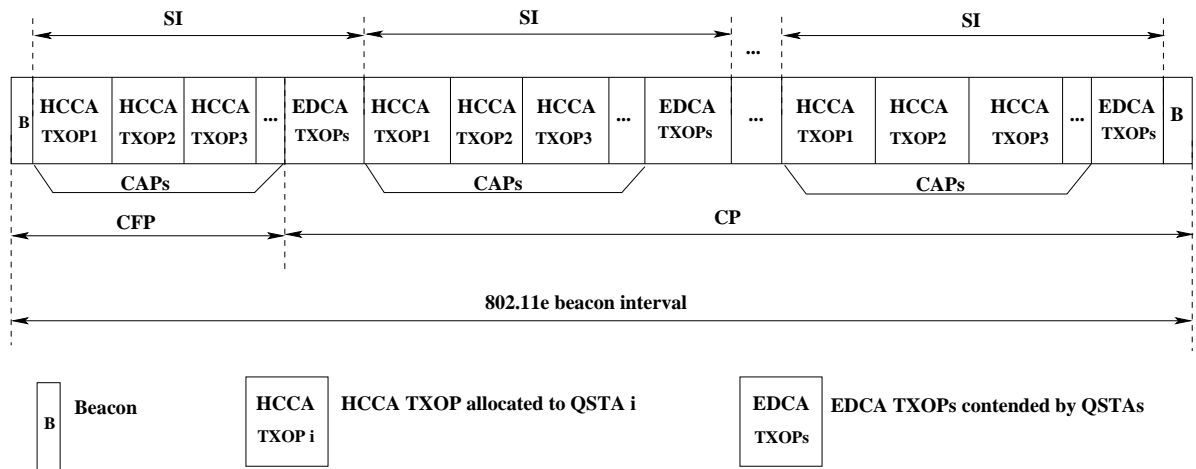


Figure 4.2: Alternation of CFP, CP and triggered CAP during an IEEE 802.11e superframe

HCCA solves the two following major problems of PCF:

- An 802.11e QSTA is not allowed to transmit a packet if the frame transmission cannot finish before the next beacon, which solves the beacon delay problem of PCF.
- A $\text{TXOP}_{\text{Limit}}$ is used to bound the transmission time of a polled QSTA.

HCCA is more flexible than PCF because the latter is only allowed in a CFP period, while a QAP can initiate HCCA whenever it wishes during the whole beacon interval. Even though PCF is allowed, the flexibility of HCCA makes PCF useless, and PCF is again defined as optional in 802.11e. To leave enough space for EDCA, the maximum duration of HCCA in a beacon interval is limited by a variable $T_{CAPLimit}$.

Before any data transmission, a traffic stream (TS) is first established, and each QSTA is allowed to have no more than eight TSs with different priorities¹. In order to initiate a TS connection, a QSTA sends a QoS request frame containing a traffic specification (TSPEC) to the QAP. A TSPEC describes the QoS requirements of a TS, such as mean/peak data rate, mean/maximum frame size, delay bound, and maximum required service interval (RSI). A maximum RSI refers to the maximum duration between the start of successive TXOPs that can be tolerated by a requesting application. Intuitively, there is a link between the maximum RSI and the delay bound for a given TS. Consequently, the 802.11e standard suggests that if both a maximum RSI and a delay bound are specified by a QSTA. The HCCA simple scheduler should only use the maximum RSI for calculating a TXOP schedule. On receiving all these QoS requests, the QAP scheduler first determines the selected service interval (SI), which should be the highest sub-multiple value of the beacon interval and also be no larger than all the maximum RSIs required by the different TSs from different QSTAs. Then an 802.11e beacon interval is divided into an integer number of SIs, and QSTAs are polled sequentially during each selected SI. In this way, all the admitted TSs should be polled once within the delay requirement of the most time-stringent TS. Lastly, the QAP scheduler computes the corresponding HCCA-TXOP values for different QSTAs by using their QoS requests in TSPECs (TXOP1, TXOP2, etc.), as shown in Figure 4.2, and allocates them to those QSTAs [90].

An HCCA admission control algorithm is also suggested in 802.11e. Using TSPEC information, the QAP calculates a ratio of the transmission time reserved for HCCA of all existing K QSTAs over an SI: $\sum_{i=1}^k \frac{TXOP_i}{SI}$. In order to decide whether or not a request from a new traffic flow can be accepted in HCCA, the QAP scheduler only needs to check if the new request $TXOP_{k+1}$ plus all the current TXOP allocations are lower than or equal to the maximum fraction of time that can be used by HCCA:

$$\frac{TXOP_{k+1}}{SI} + \sum_{i=1}^k \frac{TXOP_i}{SI} \leq \frac{T_{CAPLimit}}{T_{Beacon}} \quad (4.1)$$

where $T_{CAPLimit}$ is the maximum duration bound of HCCA, and T_{Beacon} represents the length of a beacon interval.

4.3.2 Enhanced Distributed Coordinated Access (EDCA)

The EDCA proposal of the 802.11e standard differentiate the channel access probability among different traffic sources. Packets arriving to the MAC layer (MSDUs) are mapped into four different Access Categories (AC), which represent four different level of service for the contention to the

¹Note that Traffic Stream (in HCCA) and Access Categories (in EDCA) can use different MAC queues.

shared medium. Each AC contends to the medium with the same rules of standard DCF, i.e., wait until the channel is idle for a given amount of inter frame space, and then access/retry following exponential backoff rules. The access probability differentiation is provided by:

- Different Arbitration Inter-Frame Spaces AIFS, instead of the constant DIFS
- Different values for the minimum/maximum contention windows (CW_{\min} , CW_{\max}) to be used for the backoff time extraction.

Figure 4.3 presents the method to differentiate the Access Categories (AC) according to the priority using the AIFS waiting time. Due to the service differentiation, the real-time traffic gets a higher priority in winning channel contention under 802.11e and evidently provides a better performance as compared to the basic 802.11 MAC scheme.

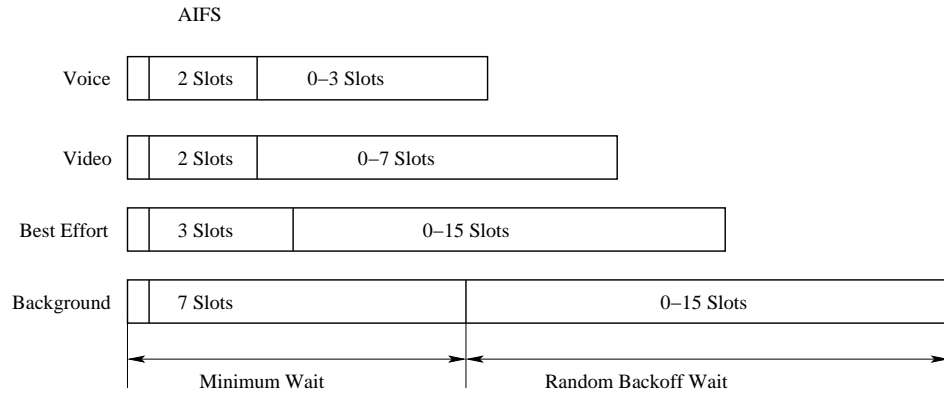


Figure 4.3: Timing diagram to support QoS according to AC in EDCA

Separate queues are maintained in each station for different ACs and each one behave as a single enhanced DCF contending entity. EDCA also specifies new channel utilization operations based on the concept of transmission opportunity (TXOP), which represents a time interval in which the station is authorized to hold the channel. Each AC is characterized by QoS parameters such as Arbitration Inter Frame Space Number (AIFSN), minimum Contention Window (CW_{\min}), maximum Contention Window (CW_{\max}), Persistent Factor (PF), Transmission Opportunity limit (TXOP limit). These parameters are transmitted at the beacon frame. The value of AIFS Time (T_{AIFS}) can be calculated using AIFSN:

$$T_{\text{AIFS}(i)} = T_{\text{SIFS}} + \text{AIFSN}(i) \times T_{\text{SLOT}} \quad (4.2)$$

where i is the AC number, T_{SIFS} is the SIFS time, and T_{SLOT} is the slot time. After each collision, the contention window is updated according to Equation 4.3:

$$CW_{j+1}(i) = \min\{ [(CW_j(i) + 1) \times \text{PF}(j)] - 1, CW_{\max}(i) \} \quad (4.3)$$

where i is the AC, j is the number of retransmission due to collision and PF is the persistent factor that exponentially increases according to the retransmission number. The back-off time T_{backoff} is

then determined using Equation 4.4:

$$T_{\text{backoff}} = \lceil \text{rand}[0, CW_{j+1}(i)] \rceil \times T_{\text{SLOT}} \quad (4.4)$$

Table 4.2 shows the default EDCA values of the channel access parameters for each AC. In each beacon frame, the QAP broadcasts the values of these parameters. The most recent EDCA parameter set element received by the stations are used to update the appropriate MAC values.

Table 4.2: EDCA default parameter set

Access Category	AC_VO (Voice)	AC_VI (Video)	AC_BE (Best Effort)	AC_BK (Background)
CW_{\min}	$\frac{CW_{\min}+1}{4} - 1$	$\frac{CW_{\min}+1}{2} - 1$	CW_{\min}	CW_{\min}
CW_{\max}	$\frac{CW_{\min}+1}{2} - 1$	CW_{\min}	CW_{\max}	CW_{\max}
AIFSN	2	2	3	7

EDCF mode can be regarded as a QoS assurance mechanism in the sense that a traffic class can statistically reduce its transmission delay by categorizing itself into a higher priority traffic class in its contention for the channel.

4.4 Related Work

There is a rich literature [92][95][96][97][100] on the capacity evaluation of WLANs in terms of VoIP simultaneous communications. In [96], the capacity of an IEEE 802.11a network was evaluated, taking into account loss at the playout buffer. The authors show that the capacity can be improved by using automatic rate selection instead of a fixed 6 Mbps physical link rate. In [95], the authors study the limitations of the 802.11 (a/b) DCF mode in supporting VoIP over a WLAN. An approximate analytical capacity model is proposed. It illustrates the importance of choosing a voice packet size that is as large as possible. With G.711 codec and DCF parameter settings, a capacity limit of 12 VoIP conversations is obtained for 802.11b (11 Mbps) WLAN. For 802.11a (54 Mbps) WLAN 56 VoIP conversations are supported. The authors also show that the actual number of VoIP calls² is further reduced by factors such as spacial distribution of the wireless stations, since the signal strength depends on the distance.

In [97], the authors propose a simple analytical upper bound of the capacity for VoIP calls in IEEE 802.11b DCF mode. The capacity of G.711 VoIP using CBR model and a 20 ms packetization interval was 12 VoIP communications. Simulations are used to show the effect of delay constraints on capacity. The analysis presented in this paper assumes that no collisions occur and that frames are always received without errors. Thus, the possibility of increase in the backoff time due to collisions is ignored in this study. Considering an arbitrary long period of time T seconds during which N VoIP conversations are in progress, the time required for the transmission of frame sequences to

²The terms calls and communications are used interchangeably.

and from the AP is given by:

$$2NRT(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}} + T_{\text{DIFS}}) \quad (4.5)$$

where R (packets/seconds) is the coding packet rate, T_{voice} is the voice data transmission time, T_{SIFS} , T_{ACK} , and T_{DIFS} are times relative to SIFS, ACK and DIFS, respectively. The time required for the AP to complete its backoff procedure is $\left[\sum_{i=1}^{i=NRT} CW_i \right] \times T_{\text{SLOT}}$, where T_{SLOT} is the slot duration, and CW_i is the number of slots picked from a uniform distribution over $(0, CW_{\min})$ for the i th transmission. For large T this expression converges to $NRT(T_{\text{SLOT}} \times \frac{CW_{\min}}{2})$. In order to support the offered load of N VoIP communications the condition 4.6 is required:

$$T \geq [2NRT(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}} + T_{\text{DIFS}})] + \left[NRT(T_{\text{SLOT}} \times \frac{CW_{\min}}{2}) \right] \quad (4.6)$$

Then, the upper bound on N is obtained according to Equation 4.7:

$$N_{\max} = \left\lfloor \frac{1}{R \left[2(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}} + T_{\text{DIFS}}) + (T_{\text{SLOT}} \times \frac{CW_{\min}}{2}) \right]} \right\rfloor \quad (4.7)$$

A similar capacity evaluation approach is introduced in [98]. In this work, the authors assume CBR VoIP sources generating one voice packet every packetization interval and that packets should be transferred right after they are generated to avoid delay. Then, the number of packets that can be sent during one packetization interval is the maximum number of VoIP calls, and the VoIP capacity can be obtained from Equation 4.8:

$$N_{\max} = \frac{T_p}{2(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}} + T_{\text{DIFS}}) + (T_{\text{SLOT}} \times \frac{CW_{\min}}{2})} \quad (4.8)$$

where T_p is the voice packetization interval. In this paper, the VoIP capacity under PCF mode is also studied. The basic assumption is that VoIP stations are polled every packetization interval in order to avoid delay in PCF mode. Thus, the CFP interval should be less than or equal to the packetization interval ($T_{\text{CFP}} \leq T_p$). Then, the capacity of VoIP in PCF mode is calculated from Equation 4.9.

$$N_{\max} = \frac{T_{\text{CFP}} - T_B - T_{\text{CE}}}{2(T_{\text{voice}} + T_{\text{SIFS}})} \quad (4.9)$$

where T_{CFP} is the CFP duration, T_B is the time needed to send a beacon frame and T_{CE} is the time for sending a CF-End frame.

In [106], simulation results show that, using HCF mode of operation, the number of simultaneous transmissions can be increased to a varying extent because of the 802.11e service differentiation. In [92], the authors analyze the capacity of a system that uses PCF for VoIP traffic, CBR and VBR models were considered. Values of 75 ms and 90 ms were used as CFP interval. The capacity for VoIP with a 90 ms CFP was 26 voice calls, but the maximum delay was 303 ms. This end-to-end delay value is not acceptable for a good voice quality.

4.5 Studied Network System

We consider an IEEE 802.11e system in infrastructure mode. We focus on the architecture presented in Figure 4.4 where a VoIP gateway acts as the conjuncture of wired Internet and wireless LAN. All voice packets go through the gateway before their transmission over the WLAN channel.

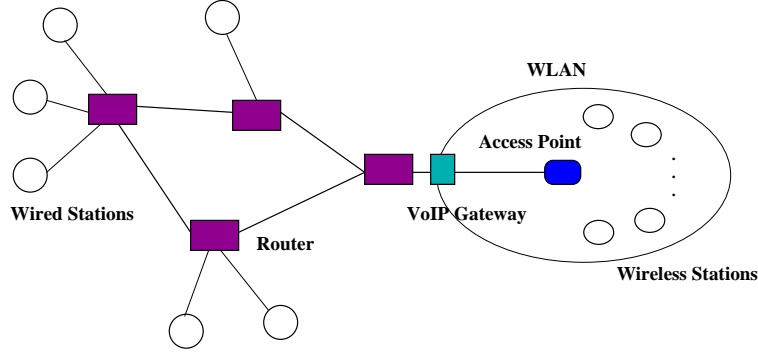


Figure 4.4: Wireless network scenario

The gateway is associated with an 802.11e QoS AP (QAP). A QAP is required to support VoIP calls between wired and wireless networks. The functionality of HC (Hybrid Coordinator) is performed at the QAP. The gateway transmits packets from the QAP and transfers them to the wired network and vice versa. The QAP has two interfaces, an 802.11 interface for the wireless traffic and an interface for the WAN traffic (wired part). We consider the case of VoIP calls established between QSTAs and wired stations. The wired network effect on VoIP call quality is not studied in this work. We assume that all VoIP packets are exchanged via the QAP. Then, for each VoIP communication, there are two streams. The uplink stream is for voice originating from the wireless station to the QAP. The downlink stream is for voice originating from the wired network and transmitted from the QAP to the QSTA. For a voice call between two QSTAs in the WLAN (intra-AP call), four streams should be considered if the communication is transmitted through the QAP. Nevertheless, depending on implementation options, the two QSTAs can directly send traffic to each other without further intervention of the QAP.

4.6 Protocol Layers and Headers Overhead

IEEE 802.11b WLAN can support data rates up to 11 Mbps. With G.729A low bit rate codec, a VoIP stream typically requires an IP bandwidth of 24 Kbps (see Table 3.1). Ideally, the number of simultaneous VoIP communications that can be supported by an 802.11b WLAN is then $\frac{11\text{Mbps}}{(24\text{Kbps} \times 2)}$, which corresponds to about 230 VoIP communications, each with two VoIP streams (downlink and uplink stream). However it turns out that WLAN can only support no more than few VoIP communications [95][97][100]. This result is mainly due to:

- The added packet-header overhead as short voice packets traverse the various layers of the

standard protocol stack.

- The inefficiency inherent in the WLAN medium-access control (MAC) protocol.

In fact, a typical VoIP packet consists of 40 bytes of IP/UDP/RTP headers and a payload ranging from 10 to 30 bytes, depending on the codec used. Therefore, the efficiency at the IP layer is already less than 50 %. Consider a VoIP packet with a 20-bytes payload. The transmission time for it at 11 Mbps is $\frac{20\text{bytes} \times 8}{(11\text{Mbps})} = 14.5\mu\text{s}$. The transmission time for the 40-byte IP/UDP/RTP header is $\frac{40\text{bytes} \times 8}{(11\text{Mbps})} = 29\mu\text{s}$.

In addition, the efficiency drop is much worse at the 802.11 MAC/physical (PHY) layers that introduce additional overhead of more than 800 μs , attributed to the physical preamble, MAC header, MAC backoff time and MAC acknowledgment (ACK) as shown in Figure 4.5.

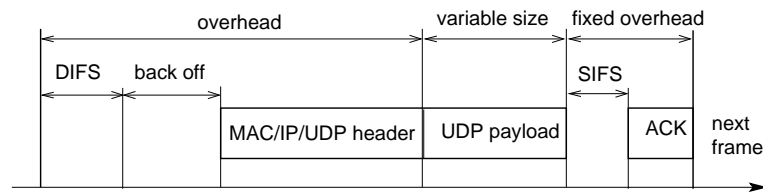


Figure 4.5: DCF overhead and data frame

For example, the IEEE 802.11b standard defines SIFS to be 10 μs . A slot time is 20 μs and the value of DIFS is 50 μs . The size of an ACK frame is 14 bytes which takes about 10 μs to be transmitted at 11 Mbps. However, each transmitted frame also needs some physical layer overhead (PLCP³ header of 48 μs and a preamble of 144 μs) which is about 192 μs . The total time to transmit an acknowledgment is then 202 μs . Furthermore, a random backoff time should be considered. The average random backoff time is 310 μs according to [91]. As a result, the overall efficiency drops to less than 3 % [99].

Despite the nominal throughput of 54 Mbps, protocol overheads are not reduced with IEEE 802.11g WLAN link (Table 4.1). For packets with large payload, higher throughput than that in 802.11b can be achieved. But, since VoIP packets typically have very small payload, the VoIP capacity increase due to higher data rate of 802.11g is limited.

4.7 VoIP Capacity in IEEE 802.11e Networks

In this Section, we describe the model that we propose to analytically compute the voice capacity of 802.11e WLANs. We analyze the capacity of Constant Bit Rate (CBR) VoIP communications, as this capacity is an upper bound on the network capacity. The WLAN network capacity in terms of simultaneous VoIP communications is defined as the maximum number of bi-directional calls that a given QAP can support while maintaining acceptable level of voice transmission quality. We

³Physical Layer Convergence Protocol

analyze numerical results obtained using the proposed capacity model while varying application level parameters (i.e., voice codecs) and MAC/PHY parameters.

4.7.1 Proposed VoIP Capacity Model

In this work, we consider the HCF (HCCA and EDCA) mode of operation of the IEEE 802.11e WLAN. In order to maximize the number of supported VoIP communications over 802.11e WLAN, we propose to transmit voice traffic under both HCCA and EDCA modes.

The maximum number of supported VoIP communications is controlled by the superframe length and the service interval SI. The service interval value can be expressed as $SI = \alpha T_{SF}$ where $\alpha \in]0, 1]$. Voice packet delay will also be determined by the SI time. If $SI < T_{SF}$, each voice call could have one or multiple TXOPs per SF and this will help providing low end-to-end packet voice delays. However, the WLAN capacity will be reduced in that case.

In this study we consider the case where a QSTA can obtain only one transmission opportunity during a SF ($\alpha = 1$). The TXOP is obtained by a successful EDCA contention (EDCA-TXOP) or by a QoS poll frame received from the QAP (HCCA-TXOP). Thus the service interval will be $SI = T_{SF}$. Interestingly, it is possible to change the superframe length since this parameter is carried in MAC beacons.

As explained in Section 4.6, protocol header overhead is significant for short voice packets, hence we recommend to send all voice data generated within SI period in one MAC packet. Thus, variable length packets will be created. In order to determine the voice packet size, assume $R_{\text{codec}}(\text{kbps})$ to be the codec bit rate, P_{min} is the packetization delay (listed in Table 4.4), SI is the service time interval. Then voice frame size is given by:

$$S_{\text{voice}} = R_{\text{codec}}(P_{\text{min}} + SI) \quad (4.10)$$

In the studied case, we consider $SI = T_{SF}$. For the actual voice frame we consider a total a header size of 74 bytes (H parameter)⁴. The physical layer overhead (PLPC header and preamble) is given by P parameter. The time required for the voice packet transmission will be:

$$T_{\text{voice}} = \frac{S_{\text{voice}} + H + P}{\rho} \quad (4.11)$$

where ρ is the physical data transmission rate of the wireless link.

For an ACK packet the transmission time is:

$$T_{\text{ACK}} = \frac{S_{\text{ACK}} + P}{\rho} \quad (4.12)$$

where S_{ACK} is the ACK packet size (14 bytes).

Then, the time required for the transmission of voice frames of a bi-directional voice call during an EDCA-TXOP will be given by Equation 4.13:

⁴34 bytes for the 802.11 MAC header, 20 bytes of IP header, 8 bytes of UDP header and 12 bytes of RTP header.

$$T_{\text{VOICE-CP}} = 2(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}} + T_{\text{AIFS}} + (T_{\text{SLOT}} \times \frac{CW_{\min}}{2})) \quad (4.13)$$

For Voice access category we consider the AIFS time $T_{\text{AIFS}} = T_{\text{SIFS}} + 2 \times T_{\text{SLOT}}$. Therefore the number of VoIP calls supported during CP under EDCA mode will be bounded by:

$$N_{\text{VOICE-CP}} = \frac{T_{\text{CP}} - T_{\text{CAP}}}{T_{\text{VOICE-CP}}} \quad (4.14)$$

where T_{CP} is the CP duration and T_{CAP} is the duration of CAPs obtained during CPs by the QAP. Similarly, we obtain the equation that determines the time required for the transmission of voice frames of a bi-directional voice call polled during an HCCA-TXOP:

$$T_{\text{VOICE-CFP}} = 2(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}}) \quad (4.15)$$

The total number of VoIP calls supported during CFP under HCCA mode and during CAPs (obtained by the QAP during the CP) will be bounded by:

$$N_{\text{VOICE-CFP}} = \frac{T_{\text{SF}} - T_{\text{CP}} - T_{\text{overhead}} + T_{\text{CAP}}}{T_{\text{VOICE-CFP}}} \quad (4.16)$$

where T_{SF} is the superframe duration, T_{CP} is the CP duration including CAPs, and T_{overhead} is the overhead time incurred when the QAP starts and terminates a CFP (see Figure 4.2). This time is given by:

$$T_{\text{overhead}} = \frac{(B + P) + (CF_{\text{end}} + P)}{\rho} \quad (4.17)$$

where B is the size of the beacon frame sent at the start of a CFP. P is the physical header size and CF_{end} is the size of the CFP-end frame (see Table 4.3). The maximum number of supported voice calls is then obtained from the addition of the two capacity bounds given by Equations 4.14 and 4.16, presenting voice calls transmitted during CP and CFP respectively. Parameters used for this modeling study are shown in Table 4.3.

4.7.2 Modeling Results for VoIP Capacity under HCCA/EDCA

We compare the capacity of CBR VoIP using G.711, G.729 and G.723.1 standard voice codecs, features of these codecs are listed in Table 4.4.

Figure 4.6 compares the quality of these codecs using their MOS as function of increasing network delay and assuming no packet loss. From this figure, we observe that MOS value of 3.6 (PSTN reference quality) can be obtained for the three codecs G.723.1, G.729 and G.711 for delay values under 170 ms, 240 ms and 325 ms respectively. For the same delay values, quality of G.729 and G.723.1 codecs is lower than that of G.711. This is because of the higher intrinsic impairment factor of low bit rate codecs (as explained in Section 2.4).

In Figure 4.7, we plot VoIP capacity in an IEEE 802.11e network (11 Mbps) using HCCA/EDCA

Table 4.3: System modeling parameters

Parameter	IEEE 802.11b	IEEE 802.11g
MAC header overhead	34 bytes	34 bytes
Physical layer overhead (P)	24 bytes	24 bytes
ACK packet size	14 bytes	14 bytes
Slot time (T_{SLOT})	20 μs	9 μs
SIFS (T_{SIFS})	10 μs	10 μs
AIFS _{voice} (SIFS + 2 \times T_{SLOT})	50 μs	28 μs
RTS	20 bytes	20 bytes
CTS	14 bytes	14 bytes
Beacon frame size (B)	40 bytes	40 bytes
CF-end frame size (CF_{end})	24 bytes	24 bytes

Table 4.4: Codec features

Codec	Bit Rate (Kbit/s)	Packetization Delay (ms)	Codec Impairment I_e
G.711	64	20	0
G.729	8	20	10
G.723.1	6.3	30	15

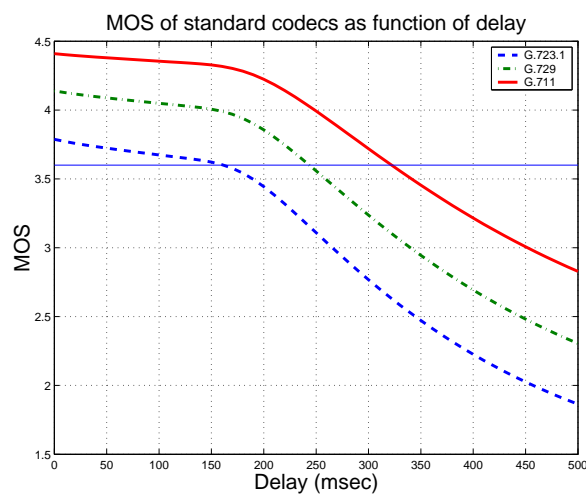


Figure 4.6: MOS of standard codecs as function of delay

mode of operation as function of increased SF duration. We compare the capacity for three standard coding rates listed in Table 4.4. This modeling result is obtained for SF duration T_{SF} which is

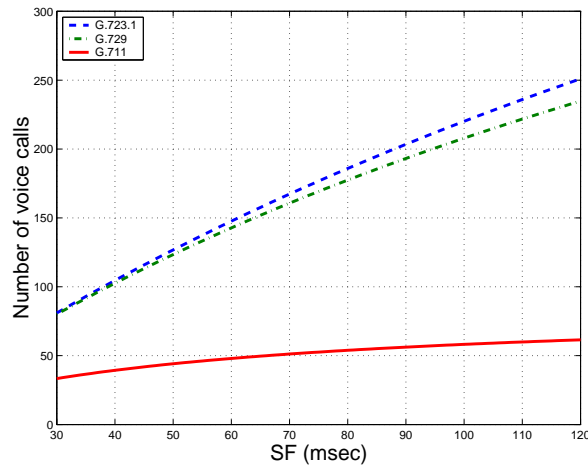


Figure 4.7: Improved maximum number of supported voice calls under EDCA/HCCA

varied from 30 ms to 120 ms. The CP duration is increased proportionally to the SF duration: $T_{CP} = 0.7 \times T_{SF}$. The CAP duration is also varied according to the CP duration: $T_{CAP} = 0.2 \times T_{CP}$. The main observation is that increasing the SF duration T_{SF} significantly improves the number of supported VoIP communications. However, the SF duration increase is at the expense of increasing voice packet delay. Hence, this should be avoided in order to not consume most of the end-to-end delay budget for voice in just one hop across the WLAN. Capacity increase is more important for low coding rates. The maximum number of VoIP calls, with a 120 ms SF, is of 61 calls when the G.711 codec is used and is much higher for G.729 and G.723.1 (234 and 251 respectively). This is because G.711 is constrained by throughput. Using G.729 and G.723.1 low bit rate codecs improves the WLAN capacity. However, this capacity improvement is at the expense of quality degradation related to intrinsic impairment factors of low bit rate codecs (see Table 2.5).

Using the same WLAN parameters, we show in Figure 4.8 (a) the VoWLAN packet delay given by: $D = P_{min} + T_{SF}$ while increasing the number of supported VoIP calls and hence increasing SF duration T_{SF} . Figure 4.8 (b) plots the MOS quality measure as function of the number of VoIP calls for no packet loss and without considering the packet delay that occurs at the wired network part. These figures illustrate an important result: large number of VoIP calls (more than 350 calls) can be supported with slower delay increase for G.723.1 codec compared to G.729 codec. This can be explained by the fact that thanks to the G.723.1 low bit rate (6.3 kbps), shorter voice frames can be generated during the same delay. Thus, for the same number of supported calls, G.729 and G.711 flows experience higher delays than G.723.1 flows. Although, G.729 represents higher intrinsic MOS quality than G.723.1, the delay increase for a large number of G.729 flows (exceeding 350 calls) causes faster quality degradation than that of G.723.1 flows.

Another observation is that when the G.711 high bit rate (64 kbps) codec is used, low VoWLAN capacity is obtained. High delays are experienced when more than 60 G.711 flows are supported

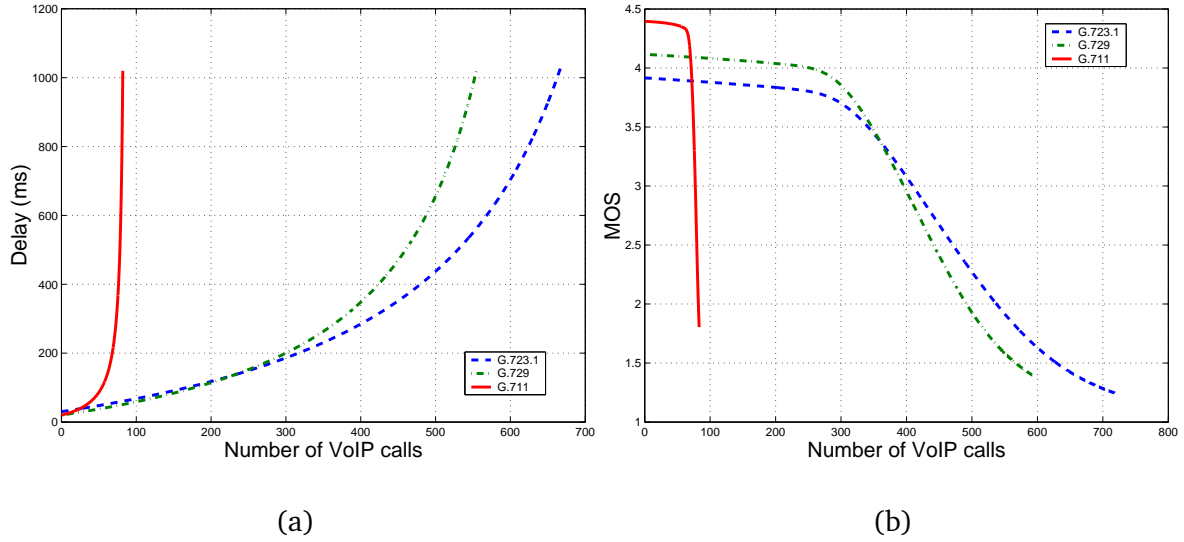


Figure 4.8: VoWLAN (a) packet delay and (b) MOS as function of the number of supported VoIP calls

(> 500 ms) and this causes unacceptable voice quality (MOS < 3). This result shows that optimal VoWLAN quality can be obtained by the use of codec adaptation mechanism that determines the voice codec to be used (by the QSTAs for uplink streams and by the VoIP gateway for down-link streams) according to the number of voice call accepted for transmission over the WLAN. For instance, under the same WLAN configuration as the one used for this result (Figure 4.8 (b)):

- G.711 should be used if the number of flows is less than 70.
- G.729 should be used if the number of flows is between 70 and 350.
- G.723.1 should be used if the number of flows is higher than 350.

In a second modeling experiment, illustrated in Figure 4.9, we present the VoIP capacity results for the three before-mentioned codecs while increasing the CP duration from 30 ms to 120 ms (SF duration is kept constant: 120 ms).

When the CP duration is increased, the VoIP capacity is considerably reduced especially for low bit rate codecs. For G.723.1 codec, the capacity drops from 287 calls (with a 30 ms CP) to 226 calls (with a 120 ms CP). This capacity decrease is mainly due to the MAC protocol overhead incurred during CPs. In CP, QSTAs have to contend for the channel and certain duration of time is wasted in contention winning and backoff window exhaustion process. Whereas in the CAP periods the Hybrid Coordinator (HC) gains control of the medium after sensing the channel idle for PIFS duration. Then the QAP polls the QSTAs which has voice traffic based on its defined polling scheme.

The capacity difference between EDCA and HCCA modes is shown in Figure 4.10.

Figure 4.10 (a) illustrates the VoIP capacity in a 802.11e under exclusive HCCA mode as function of increased SF duration. Figure 4.10 (b) illustrates the capacity under exclusive EDCA mode. HCCA mode achieves larger increase in the VoIP capacity (153 G.723.1 calls for a 120 ms SF) than EDCA

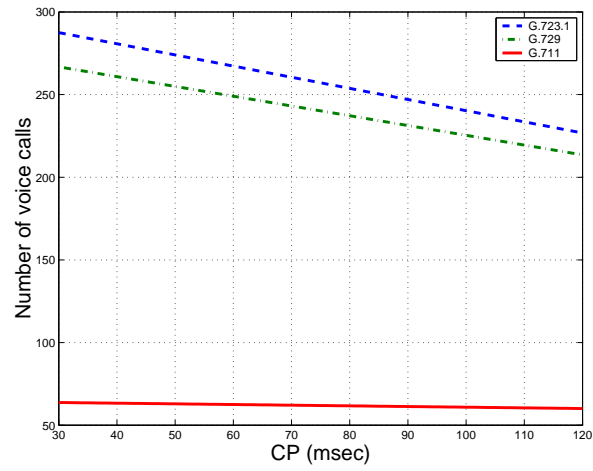


Figure 4.9: Decreased VoIP capacity under EDCA/HCCA when CP duration is increased

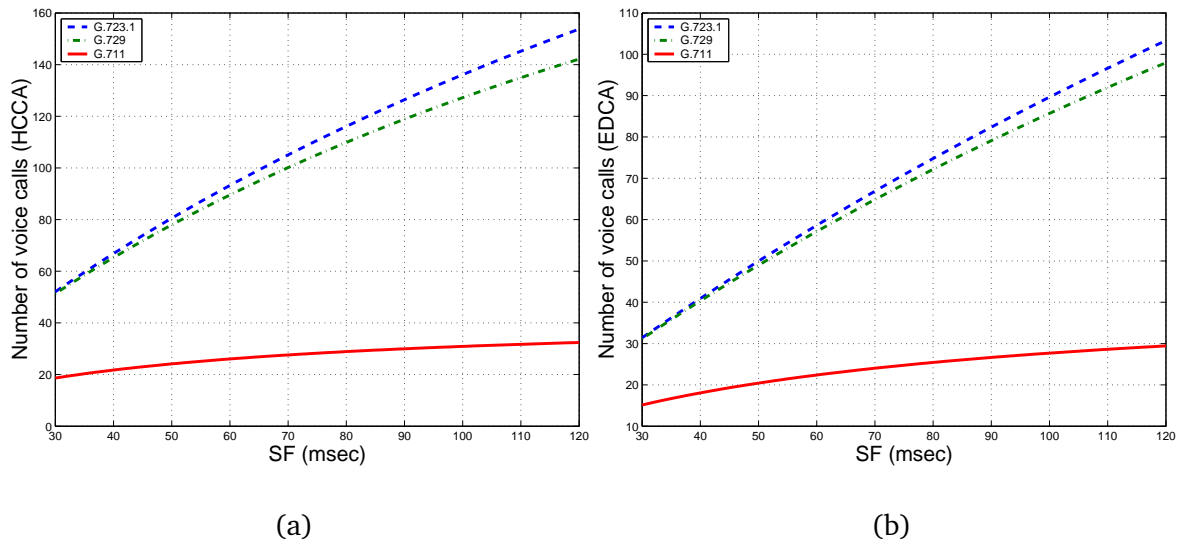


Figure 4.10: Maximum number of supported voice calls under (a) HCCA/(b) EDCA

mode (103 G.723.1 calls for a 120 ms SF). This result is due to the protocol efficiency and the low overhead of HCCA mode.

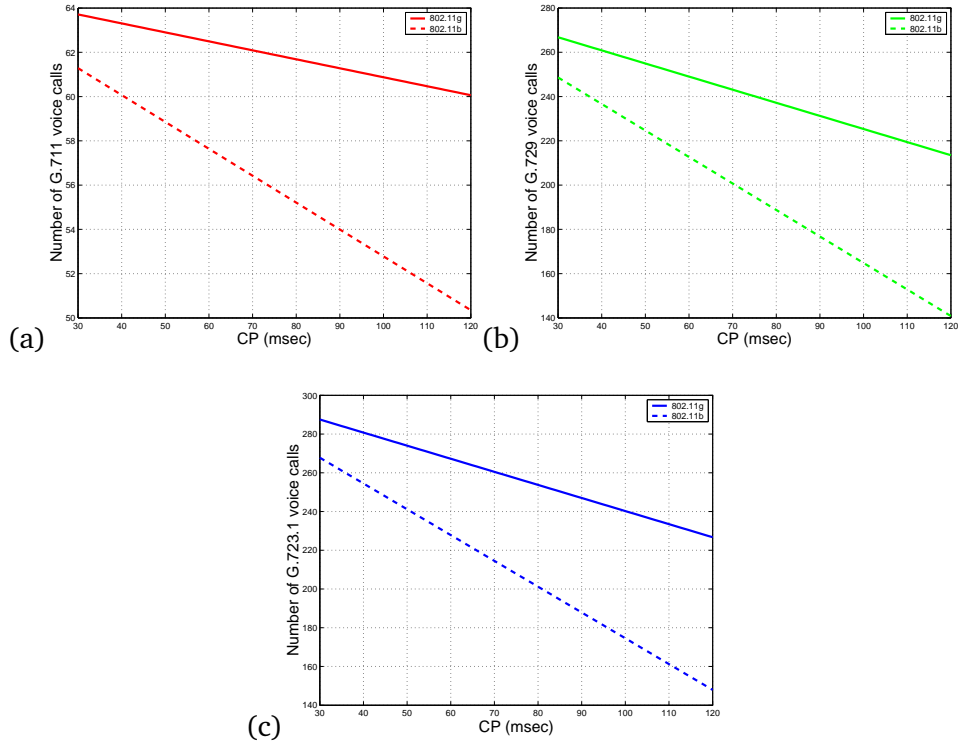


Figure 4.11: Comparison of VoIP capacity under 802.11b/g for (a) G.711 (b) G.729 and (c) G.723.1

In Figure 4.11, we compare VoIP capacity of 802.11b and 802.11g WLAN for the same physical transmission rate (11 Mbps). In this modeling experiment, we increase the CP duration from 30 ms to 120 ms (SF duration is kept constant: 120 ms). An important observation is that with 802.11b standard the capacity drops more significantly (from 267 to 147 G.723.1 calls for SF duration of 30 ms and 120 ms respectively) than the case of 802.11g standard (from 287 to 226 G.723.1 calls for SF duration of 30 ms and 120 ms respectively). This result can be explained by the MAC overheads used for the contention-based channel access mechanism (i.e., AIFS times) which are higher for 802.11b than for 802.11g.

Figure 4.12 illustrates the effect of the physical transmission rate on the VoIP capacity for 802.11b/802.11g WLAN. This modeling result is obtained for a constant SF duration $T_{SF} = 120$ ms, a CP duration $T_{CP} = 0.7 \times T_{SF}$, and CAP duration $T_{CAP} = 0.2 \times T_{CP}$.

Table 4.5 and 4.6 summarize numerical results of the capacity according to standard 802.11b/802.11g physical data rates. Recent technologies such as IEEE 802.11g with high data rates (up to 54 Mb/s) promise important VoIP capacity when the EDCA/HCCA mode is used (up to 228 G.711 VoIP calls, 561 G.719 VoIP calls and 590 G.723.1 VoIP calls).

For the same physical data transmission rate (i.e., 11 Mbps), VoIP capacity achieved by 802.11b WLAN (e.g., 200 G.723.1 calls) is lower than that of 802.11g (251 G.723.1 calls). This is because of the difference in MAC parameters (i.e., Slot time, AIFS time) between 802.11b and 802.11g

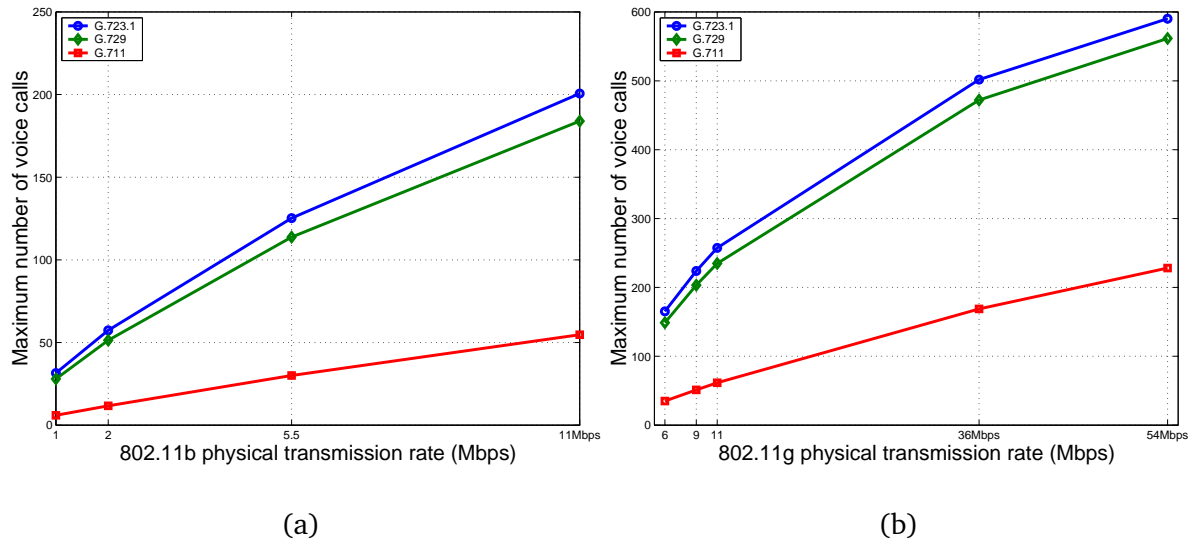


Figure 4.12: Maximum number of supported voice calls under HCCA/EDCA as function of increasing 802.11b/802.11g physical transmission rate

Table 4.5: Maximum number of VoIP calls for standard voice for different 802.11b channel transmission rates

Physical Rate	G.723.1 (6.3kb/s)	G.729 (8kb/s)	G.711 (64kb/s)
1 Mbps	31	27	5
2 Mbps	57	51	11
5.5 Mbps	125	113	30
11 Mbps	200	183	54

Table 4.6: Maximum number of VoIP calls for standard voice codecs as function of 802.11g data transmission rates

Physical Rate	G.723.1 (6.3kb/s)	G.729 (8kb/s)	G.711 (64kb/s)
6 Mbps	165	148	34
9 Mbps	223	203	51
11 Mbps	251	234	61
36 Mbps	501	472	168
54 Mbps	590	561	228

standards, as shown in Table 4.3.

4.7.3 Effect of RTS/CTS protection mode on VoIP Capacity

QSTAs may send an RTS as the first frame of any frame exchange sequence for which improved protection is desired. As we would expect, the complete RTS/CTS cycle reduces the capacity of 802.11 WLAN since the time needed for sending voice frame generated during the CP will be increased by RTS/CTS and SIFS times:

$$T_{\text{VOICE-CP}} = 2(T_{\text{voice}} + T_{\text{SIFS}} + T_{\text{ACK}} + T_{\text{AIFS}} + (T_{\text{SLOT}} \times \frac{CW_{\min}}{2})) + 2(T_{\text{RTS}} + T_{\text{CTS}} + 2 T_{\text{SIFS}}) \quad (4.18)$$

where $T_{\text{RTS}} = \frac{\text{RTS}}{\rho}$ and $T_{\text{CTS}} = \frac{\text{CTS}}{\rho}$.

RTS is the RTS frame size (20 bytes), CTS is the CTS frame size (14 bytes) and ρ is the data transmission rate. Figure 4.13 illustrates the effect of RTS/CTS mechanism on VoIP capacity under

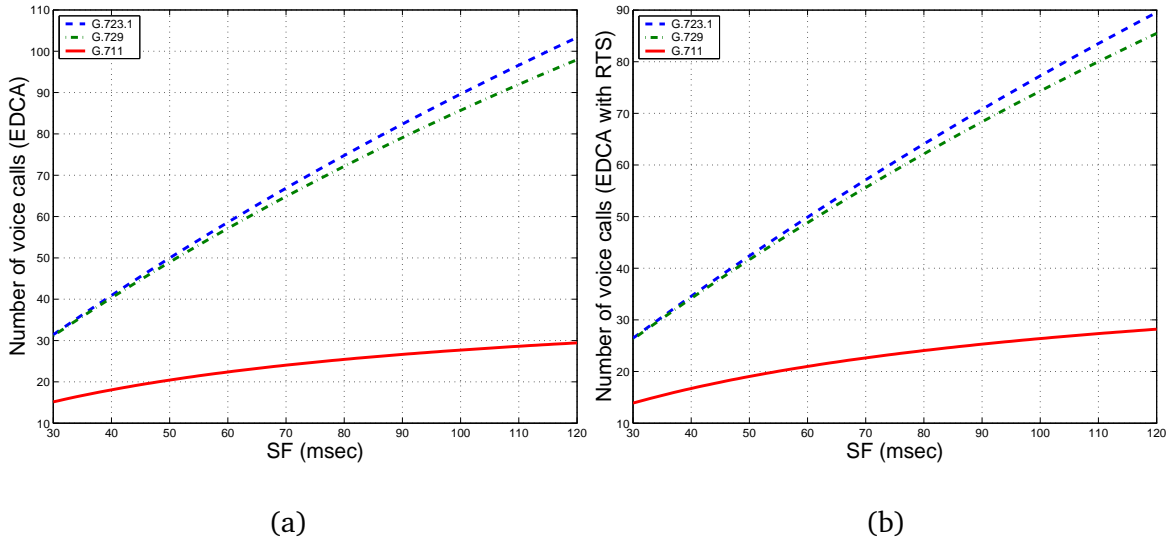


Figure 4.13: Comparison of the maximum number of supported voice calls under EDCA when RTS/CTS option is (a) disabled/ (b) enabled

EDCA. These results are obtained for $T_{\text{CP}} = 0.5 \times T_{\text{SF}}$. We summarize significant capacity modeling results for $T_{\text{SF}} = 120$ ms in Table 4.7.

Table 4.7: Effect of RTS/CTS on VoIP capacity under EDCA for $T_{\text{SF}} = 120$ ms

RTS/CTS	G.723.1	G.729	G.711
Disabled	103	97	29
Enabled	89	85	28
Reduction (%)	13.59	12.37	3.4

When RTS/CTS option is used the reduction of the maximum number of VoIP communications is more significant for lower bit rate codecs (13.59% for G.723.1, 12.37% for G.729, vs. 3.4% for G.711). This can be explained by the fact that RTS/CTS overhead is higher for small voice frames. Small voice frames encounter less collisions than larger ones, thus RTS/CTS option should be enabled only to protect large voice frames against wireless channel errors.

4.7.4 VoIP Capacity with VBR Sources

VBR encoding (i.e., using silence suppression scheme) can reduce VoIP traffic so that the capacity for VBR VoIP (see Annex A) will be larger in WLAN. In Brady's model [59], the assumed mean ON time is 1 sec and the mean off time is 1.35 sec. Thus, on average the traffic load of VBR is $\eta = \frac{\text{ON}}{\text{ON}+\text{OFF}} = 42.5\%$ of the traffic load of CBR. Ideally, the VBR VoIP capacity is given by [99]:

$$C_{\text{VBR}} = C_{\text{CBR}}/\eta \quad (4.19)$$

where C_{CBR} is the capacity for CBR source and C_{VBR} is the capacity for VBR source.

Nevertheless, if the VBR voice peak rate (i.e., rate during ON period) is used by the QAP for HCCA scheduling, voice will receive a large enough TXOP allocation for transmission. But, the channel will be underutilized and a smaller number of VBR voice flows can be admitted than in the case where the mean rates are used.

Efficient transmission of VBR VoIP traffic can be achieved using statistical data of VoIP traffic. For instance, one approach [93] [98] is to estimate the duration of the next silence period precisely, then the QSTA can be added to the polling list (during CFP) before it starts to send voice packets. However, since the talk-spurts are statistically independent, it is very difficult to estimate the duration of the next silence period precisely. If a QSTA is added into the polling list too early, then the CF-Polls are wasted until it starts to send voice packets. If a QSTA is added too late, then the voice packets should be sent in the CP until the QSTA is added in the polling list.

4.8 Conclusions

In this chapter, we have proposed an analytical model for VoIP capacity in IEEE 802.11e WLAN. Modeling results show that the EDCA/HCCA mode of operation enhances the capacity of the network in terms of the number of simultaneous supported VoIP calls. Nevertheless, EDCA mode provides lower VoIP capacity than HCCA because of MAC protocol overhead related to the contention mechanism. We compared the capacity of CBR VoIP using G.711, G.729 and G.723.1 standard voice codecs. Low bit rate codecs (i.e., G.729 and G.723.1) significantly increase VoIP capacity in comparison to G.711 codec. We observed that the superframe (SF) size is a trade-off between the WLAN capacity and the voice packet delay. Although larger SF improves VoIP capacity, it consumes most of the end-to-end delay budget for voice in the WLAN network hop. We show that optimal VoWLAN quality can be obtained by the use of codec adaptation mechanism that determines the voice codec to be used according to the number of supported voice calls. Modeling results also show that

RTS/CTS mechanism reduces the VoIP capacity especially for small voice frames. Hence, RTS/CTS option should be enabled only to protect large voice frames against wireless channel errors.

Chapter 5

Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks

Summary

The Internet today is no longer completely wired, especially with Wireless LANs (WLANs) getting appended to it. In WLANs, the physical layer's error rate is larger than that of wired networks and the challenges of the wireless channel make physical layer data rate improvements difficult to achieve. In order to deploy VoIP applications over heterogeneous wired/wireless Internet environments, efficient congestion control algorithms are required. In this chapter, we present an adaptive architecture for the transport of VoIP traffic over such environments. This architecture uses a VoIP gateway associated with an 802.11e QoS enhanced access point (QAP) to transcode voice flows based on the estimation of wireless channel congestion state. The proposed adaptation mechanism dynamically adapts codec rate using a delay-based congestion avoidance technique and allows the differentiation between congestion and wireless losses. A case study presenting the results relative to an adaptive system transmitting at bit rates typical of G.711 PCM (64 kbit/s) and G.726 AD-PCM (40, 32, 24 and 16 kbit/s) voice coding standards illustrates the performance of the proposed framework. We perform extensive simulations to compare the performance between our adaptive audio rate control and TFRC mechanism. The results show that the proposed mechanism achieves better voice transmission performance, especially when the number of stations is fairly large.

5.1 Introduction

The Internet heterogeneity is increasing due to the fast deployment of wireless local area networks (WLANs). WLAN hold the promise of providing unprecedented mobility, flexibility and scalability than its wired counterpart. At the same time Wireless VoIP, typically over 802.11 WLAN, is becoming increasingly popular, but even further elevates the challenges of delay and loss reduction. Degradation of speech quality caused by packet delay and loss of voice packets is still one of critical technical barriers of the VoIP system. Furthermore, WLANs will need to support a large number of concurrent VoIP communications since VoIP is spreading rapidly especially in public spaces. There is a serious concern from the operators side as to offer at least the current “circuit switched” quality for future VoIP communications. In order for this to become reality, a lot of issues related to VoIP transmission over heterogeneous wired/wireless networks must be solved.

In WLAN environments bandwidth is scarce, channel conditions are varying and high loss rates may occur. Even if a lot of voice codecs can tolerate some small loss without severe degradation, voice traffic has unacceptable performance if long delays are incurred. Moreover, the original IEEE 802.11 WLAN standard [85] has been mainly designed for data applications. Two different channel access mechanisms are specified in the 802.11 standard, namely, the contention-based DCF and the polling-based PCF access mechanisms. While DCF and PCF may provide satisfactory performance in delivering best-effort traffic, they lack the support for QoS requirements posed by real time traffic such as VoIP. These requirements make the DCF scheme an infeasible option to support QoS for VoIP traffic. Apart from these limitations, a typical WLAN with 11 Mbps bandwidth could only support a limited number of VoIP connections in DCF mode. In addition, studies on carrying VoIP over WLAN in PCF mode in [94] found that when the number of stations in a basic service set (BSS) is large, the polling overhead is high and results in excessive end-to-end delay and that VoIP still get poor performance under heavy load conditions. The medium access control (MAC) layer of the emerging IEEE 802.11e [86] standard tries to support QoS in 802.11 wireless networks using a Hybrid Coordination Function (HCF) that provides stations with prioritized and parametrized QoS access to the wireless medium (as described in Chapter 4). The HCF scheduler proposed in 802.11e standard considers the QoS requirements of flows by allocating transmission time to stations based on their mean sending rate and mean frame size. In this work, we investigate the performance limitations in the case of a large number of VoIP flows transmitted over an IEEE 802.11e WLAN. We specifically address the problem of long distance VoIP transport over heterogeneous wired/wireless networks. In the studied case we consider VoIP traffic transmitted from a wired Internet part through a last-hop wireless link that represents the bandwidth bottleneck. All voice traffic needs to be routed through an 802.11e QAP (QoS-enhanced Access Point). The QAP becomes heavily loaded, especially when the number of active stations is fairly large and this results in different types of audio performance degradation (loss due to congestion, loss due to bit errors at the link layer and packet delay). We show through simulations the performance of VoIP according to the number of wireless stations in a BSS. We propose an architecture that is based on a VoIP gateway for inter-working the wired and wireless networks. The gateway communicates with a QAP in order to adapt coding rate of voice

flows according to the wireless channel conditions. Simulations show that our adaptive audio rate control outperforms TFRC mechanism. The chapter is organized as follows: Section 5.2 describes related work on rate and loss control for multimedia applications. Section 5.3 states the problem. In Section 5.4, we advance the proposed architecture and we explain our adaptation algorithm. We discuss simulation results in Section 5.5. Finally, Section 5.6 concludes the chapter.

5.2 Related Work

In this section we describe related work on rate and loss control for multimedia applications over heterogeneous wired/wireless network environments.

5.2.1 Rate Control

Rate control is an important issue for both wired and wireless multimedia applications using unresponsive transport protocols (i.e., UDP and RTP). A proper form of congestion control is needed in order for these applications to share congested bandwidth fairly with each other and with TCP-based applications. Many schemes were developed based on TCP-friendly control mechanisms. These mechanisms can be classified into three main categories: equation-based mechanisms, window-based mechanisms and additive increase, multiplicative decrease (AIMD) mechanisms. Equation-based rate control [78] [79] is a widely popular rate control scheme over wired networks, also known as TCP-Friendly Rate Control (TFRC). In this scheme, the sender uses an equation characterizing the allowed sending rate of a TCP connection as a function of the RTT and packet loss rate, and adjusts its sending rate according to those measured parameters. A key issue is then to choose a reliable characterization of TCP throughput. A formulation of the TCP response function was derived in [82], it states that the average throughput of a TCP connection is given by:

$$T(\text{Bytes/sec}) = \frac{S}{RTT\sqrt{\frac{2p}{3}} + RTO(3\sqrt{\frac{3p}{8}})p(1 + 32p^2)} \quad (5.1)$$

Equation 5.1 that we have also used in Voice-TFCC rate estimation described in Chapter 3, roughly describes TCP's sending rate as a function of the frequency of loss indication p , round trip time RTT and packet size S . This equation reflects TCP's retransmit timeout behavior (RTO), as this dominates TCP throughput at higher loss rates. In the scheme proposed in [82] the receiver acknowledges each packet, and at fixed time intervals the sender estimates the packet loss rate experienced during the previous interval and updates the sending rate using Equation 5.1. This scheme updates the sending rate at fixed time intervals, hence it is suitable for use with multimedia applications. Nevertheless, it has the disadvantage of having a poor transient response at small time-scales [12].

In [84], Floyd et al. developed the TFRC (TCP-Friendly Rate Control) protocol. TFRC estimates the recent *loss event* rate of a connection at the receiver. A *loss event* consists of one or more packets dropped within a single RTT. The algorithm used for calculating the loss event rate (average loss

interval) offers a good responsiveness to changes in congestion while avoiding abrupt reductions of the sending rate in response to a single loss.

To behave in a TCP-friendly manner, the sender adapts according to an equation that models TCP response function in steady-state. The main advantages of TFRC are: first it does not cause network instability, thus avoiding congestion collapse. Second, it is fair to TCP flows. Third, the TFRC's rate fluctuation is significantly lower than that of the standard TCP congestion control algorithm, making it more appropriate for real-time applications that require a smooth congestion control and a constant quality. Window-based mechanisms such as TEAR [105] maintain a congestion window to control the transmission of packets. TEAR shifts TCP emulation to the receiver and uses a sliding window to smooth sending rates. The main disadvantage of this type of mechanisms is the lack of flexibility related to the TCP window dynamics [12]. Unlike window-based mechanisms, AIMD mechanisms [103][104] are rate-based congestion control mechanisms that are not applied to a congestion window. The Rate Adaptation Protocol [103] implements an AIMD algorithm based on regular acknowledgments sent by the receiver. In [104], the authors propose an end-to-end rate adaptation scheme that adjusts the transmission rate of multimedia applications to the congestion level of the network. Based on the estimation of the loss rate and the RTT obtained from the regular information of RTCP [9] reports, the sender increases the transmission rate during under-load periods and reduces this rate during congestion periods, while avoiding an aggressive adaptation behavior.

Although TCP-friendly rate control mechanisms provide relatively smooth congestion control for real-time traffic, they are more appropriate for use over wired IP networks. For multimedia applications over wireless, packets can be corrupted by wireless channel errors at the physical layer and thus TFRC cannot distinguish between packet loss due to congestion and that due to bit errors. TFRC, designed to deal with congestion in wired networks, treats any loss as a sign of congestion.

The second limitation of TFRC mechanisms is that they are originally designed for applications that use fixed packet size, and vary their sending rate in packets per second in response to congestion. Hence, they should not be used for applications that vary their packet size instead of their packet rate in response to congestion [83]. Varying the packet size during the time interval between two estimations of the sending rate distorts packet-based measurement of loss event.

5.2.2 Loss Recovery

In some situations, using rate control alone does not solve the performance degradation. Such situations may be short-term transient congestion, congestion caused by others' traffic or residual bit errors caused by a noisy wireless link. Forward Error Correction (FEC) has been one of the main methods used to protect against packet loss over packet switched networks and also to improve the quality of noisy transmission wireless links. The amount of FEC information should be tuned according to the characteristics of the loss process in order not to increase bandwidth requirement when the channel is loss free and also not to increase the end-to-end delay since the destination typically has to wait longer to decode the original data packet [11]. The rate/error control advocated in [11] is based on an optimization problem. This approach lacks taking delay into consideration.

In [12] an adaptive error control scheme for real-time audio over the Internet is developed. In this work the FEC scheme is selected according to its impact on the end-to-end delay using an utility function for assessing the perceived audio quality that consider the effect of the end-to-end delay. These error control schemes were designed to resolve audio packet loss over the wired IP networks; the packet loss process is different in wireless environments where loss may occur due to congestion or to residual bit errors at the link layer.

5.2.3 Wireless and Congestion Loss Differentiation

The aim of loss differentiation schemes is to differentiate between wireless and congestion losses. These mechanisms are either end-to-end or explicit. Explicit schemes are those that make use of agents deployed on intermediate network nodes. End-to-end schemes try to differentiate losses at receiver without making use of any intermediate nodes.

Explicit loss differentiation was proposed in the context of TCP flows by the use of specific agents [108] for first hop/last hop wireless topology encountered in mobile networks. The agent maintains a cache of TCP packets that have been sent from a fixed host but not acknowledged by a mobile host. It detects a loss of packet on the wireless link by seeing duplicate ACKs or by a timeout based on locally maintained timers. When it does, it retransmits the lost packet if it has been cached, since these packets are lost on the wireless link. Note that this mechanism was proposed for reliable protocols (i.e., TCP) since it retransmits the lost packets being cached. Thus, it is not suited for real-time flows since these flows are carried by unreliable protocols (i.e., RTP/UDP). In addition, for real-time applications, local retransmissions also may not help as retransmission delay may sometimes be unacceptable (e.g., for interactive VoIP communications).

For real-time flows, agents which send information either to sender or receiver about wireless and congestion losses, by either marking packets or generating new packets, are required. Explicit Congestion Notification (ECN) [109] is also a possible candidate for real-time flows. ECN is used by routers to signal congestion in the network by marking IP packets. When TFRC is used in conjunction with ECN, the losses detected when marked IP packets are received are considered congestion losses and those detected when unmarked IP packets are received, are considered to be wireless losses. However, using ECN can be inaccurate. For example, for a continuous bunch of wireless and congestion losses, ECN would classify all of them to be congestion losses.

End-to-end schemes work at the transport level in the receiver. They make use of facts such as the time taken by the packet to travel through the network and packet inter-arrival time. In [110], the authors propose a scheme based on packet inter-arrival time suitable for last hop wireless bottleneck link. This scheme limitation is that the wireless channel is generally varying and the packet delay over the wireless link keeps changing, causing misclassification. Another approach reported in [111], is based on the fact that when delay is plotted against time at the receiver, one can observe delay spikes during congestion. Thus, losses during spikes are classified as congestion losses, and otherwise as wireless losses.

End-to-end statistics can be used to help detecting congestion when packet loss happens. For example, by examining trends in the one-way delay variation, loss could be interpreted as a sign of

congestion if this delay is increasing, and as a sign of wireless channel error otherwise [101]. The scheme presented in [115] combines packet inter-arrival times and relative one-way delay to differentiate between congestion and wireless packet losses. This scheme is based on the observation that the one-way delay increases monotonically when there is congestion and that the inter-arrival time is expected to increase if there is wireless channel packet loss. End-to-end loss differentiation schemes generally have topological constraints and need to be tuned for specific topologies.

Loss differentiation algorithms can be combined with TFRC to achieve a rate control over heterogeneous Internet environments. Although some of these do improve TFRC throughput, they all suffer from significant misclassification. Therefore, congestion losses in the network may increase and TCP-friendliness of flows may decrease.

5.3 Problem Statement

One of the most sensitive case of multimedia traffic is VoIP. In particular the delay is most critical in VoIP applications. It is recognized that the end-to-end delay has a great impact on the perceived quality of interactive conversations with a threshold effect around 150 ms [3]. For intra-continental calls, the packet delay is on the order of 30 ms and for inter-continental calls the delay can be as large as 100 ms [107]. While reducing the effect of a small jitter can be realized by a playout buffer introduced at the receiver, the avoidance of a high jitter/delay is much more complex. Especially retransmissions and contention-based medium access schemes are accountable for high delays and jitters. In this work, we consider the case of VoIP calls that are transmitted over heterogeneous wired/wireless networks, we assume that the wired Internet part is error-free and congestion-free and that the bandwidth bottleneck is the last-hop wireless link (Figure 5.1). In this case, all the voice traffic needs to be routed through the 802.11e QAP (the “bridge” between the wired and wireless networks). Hence, the QAP becomes heavily loaded, especially when the number of active stations is fairly large. Moreover, 802.11e EDCF mode grants different priorities to specific traffic classes (i.e., latency sensitive traffic) but not specific nodes. Therefore, VoIP packets may be queued at the QAP if it cannot gain TXOPs from the competition with other nodes, and will become a bottleneck in the network and this will cause additional delays.

Three different types of degradation may occur in the last-hop wireless link: packet loss due to congestion, delay due to congestion and packet loss due to wireless channel errors. Although a lot of voice codecs can tolerate some small loss without severe degradation, most of them operate under preset schemes for data and channel code rates making them vulnerable to the varying conditions on wired and wireless IP-based hops [102]. Some kind of adaptation is therefore needed to dynamically adapt the codec bit rate to the changing wireless network conditions so as to preserve acceptable levels of reliability and quality.

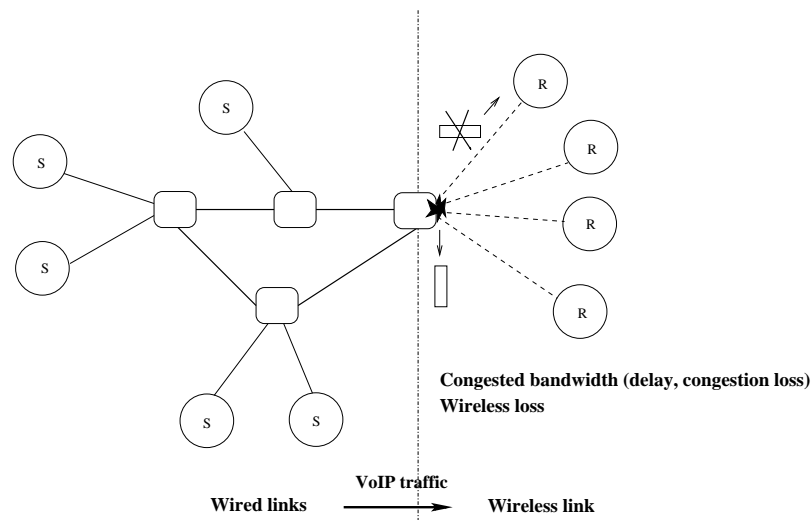


Figure 5.1: Voice traffic transmitted from wired network through a last hop congested wireless link

5.4 Proposed Architecture and Rate Control Mechanism

We present in this section, the network architecture and the adaptive rate and loss control mechanism that we propose to enhance VoIP performance over heterogeneous wired/wireless 802.11e network environments. The adaptive rate control mechanism is inspired from TCP Vegas congestion avoidance algorithm that we describe below.

Due to the heterogeneity growth of the Internet, end-to-end solutions are getting less effective. For that reason, we base our architectural proposal on intermediary nodes (i.e., VoIP gateways) interconnecting wired/wireless network parts. The VoIP Gateway is used to gather report feedbacks from VoIP end-points and to assess the WLAN congestion state. Then adaptive control mechanism is executed on VoIP flows.

5.4.1 VoIP Gateway Inter-working Wired and Wireless Network

The proposed architecture (Figure 5.2) uses a VoIP gateway located at the edge of the wired network and the WLAN, to transcode voice flows before their transmissions over the wireless channel. VoIP communications can be made between WLAN users and wired, cellular, or Internet VoIP applications. The gateway is associated with an 802.11e QAP. A QAP is required to support VoIP calls between wired and wireless networks. In such a situation, the functionality of HC (Hybrid Coordinator) is performed at the QAP. The QAP may gain high priority to access the channel by piggybacking data packets on the QoS-Poll packets or the QoS ACK packets, and thus speeds up dispatching packets from wired networks. The instantaneous bit rate is determined by an adaptation algorithm (described in Section 5.4.3) based on the estimation of wireless channel congestion state. Congestion control information can be obtained through RTCP reports sent back to sources via the HC.

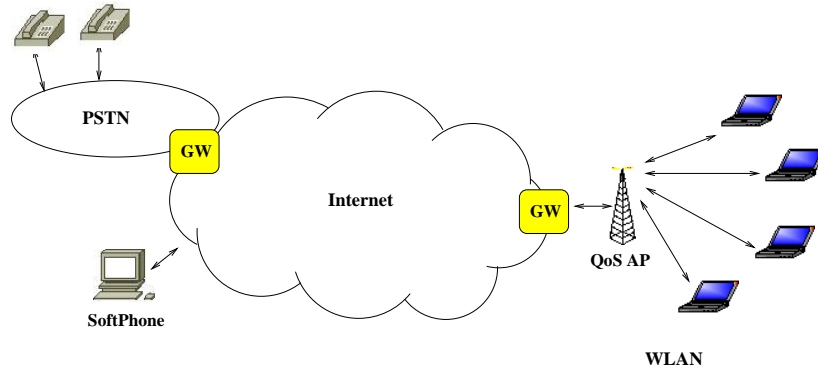


Figure 5.2: VoIP gateway at the edge of wired and wireless network

5.4.2 TCP Vegas Congestion Control Mechanism

In order to efficiently utilize network bandwidth, TCP Vegas [112] is based on a congestion avoidance technique that does not use packet loss as congestion indication. TCP Vegas evaluates the available network bandwidth using the difference between the expected and actual rates. The basic idea is that the further away the actual rate gets from the expected throughput, the more congested is the network, which implies that the sending rate should be reduced. The threshold β triggers this decrease. On the other hand, when the actual and expected throughputs are close, the connection is in danger of not utilizing the available bandwidth. The α parameter triggers this increase. TCP Vegas algorithm can be summarized as follows. Once per round trip time:

- TCP Vegas computes the expected throughput:

$$\text{Expected} = \text{cwnd} / \text{baseRTT}$$

where cwnd is the current window size and baseRTT is the minimum of all measured RTTs.

- Vegas calculates the current Actual sending rate by using the actual round trip time:

$$\text{Actual} = \text{cwnd} / \text{RTT}$$

where RTT is the observed round trip time of a packet.

- Vegas computes the extra packets in network buffers by multiplying the rate difference by baseRTT [113]:

$$\text{Diff} = (\text{Expected} - \text{Actual}) * \text{baseRTT} \quad (5.2)$$

- Finally, Vegas updates the TCP window size as follows:

$$\text{cwnd} = \begin{cases} \text{cwnd} + 1 & \text{if } \text{Diff} < \alpha \\ \text{cwnd} & \text{if } \alpha \leq \text{Diff} \leq \beta \\ \text{cwnd} - 1 & \text{if } \text{Diff} > \beta \end{cases} \quad (5.3)$$

TCP Vegas controls its window size to keep the measured extra packets within the limits $[\alpha.. \beta]$. The reason behind this is that TCP Vegas tries to detect and utilize the extra bandwidth whenever it becomes available without congesting the network. Typical values of α and β are 1 and 3 or 2 and 4 [112].

5.4.3 Vegas-Like Audio Rate and Loss Control Mechanism

We propose a rate control mechanism that is based on a TCP Vegas-like congestion avoidance technique for the rate and loss control of VoIP flows over the WLAN. The rate of the audio codec used for transcoding the voice flow at the VoIP gateway is varied according to the RTT measured between the QAP and wireless stations. In case of packet loss, a delay-based loss predictor is used to determine the type of loss and apply the appropriate strategy depending on whether packet losses are due to network congestion (transcode the voice flow using a lower audio codec bit rate) or wireless link errors (increase robustness by adding application layer FEC). The source and channel adaptation algorithm residing at the gateway will converge to the available bandwidth in the WLAN while attempting to optimize overall call quality of several simultaneous voice communications. Figure 5.3 illustrates the flow-chart of the proposed mechanism. The input of the algorithm is the

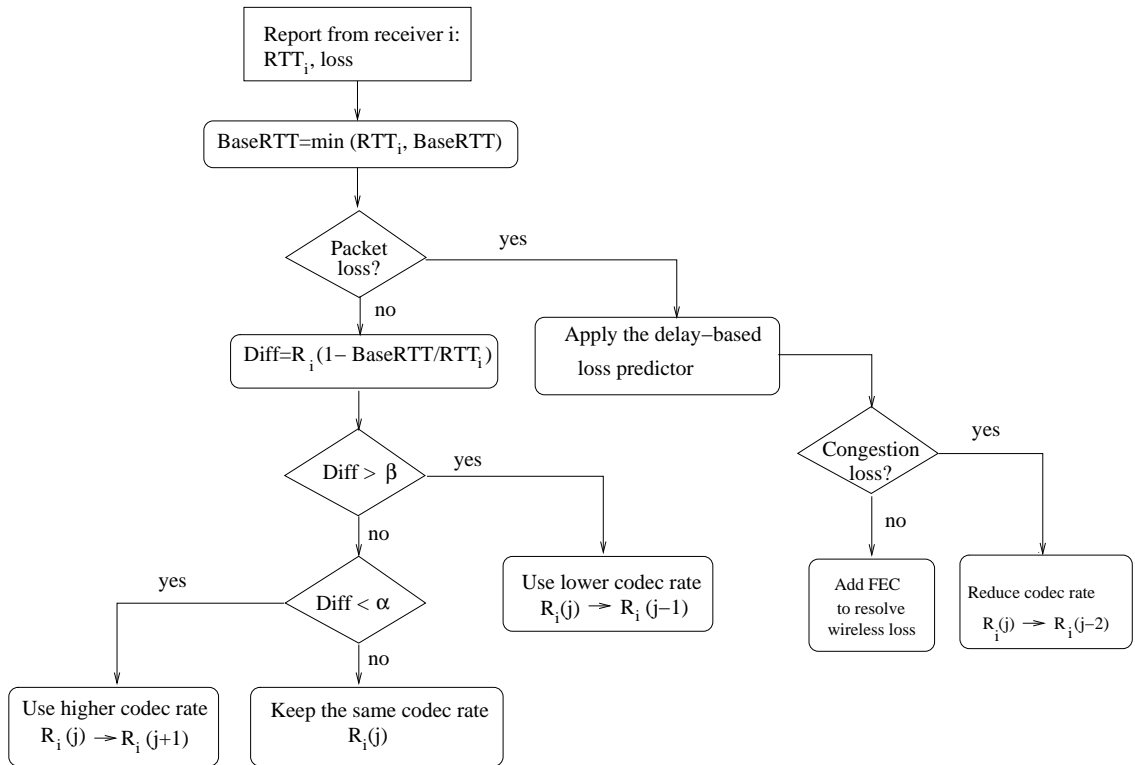


Figure 5.3: Flow-chart for the Vegas-like audio rate control

estimation of current WLAN congestion state given by delay and loss parameters. The basic idea of our Vegas-like audio rate control algorithm is to adapt the audio codec rate by varying audio packet size to avoid congestion, and this unlike TFRC mechanism that uses fixed size packets and

varies the sending rate in packets/sec. The VoIP gateway keeps track of the *BaseRTT* defined as the minimum of all RTTs measured on the WLAN using RTCP receiver reports. When a receiver report related to the voice flow i is received at the gateway, the *Expected Audio Data* and the *Actual Audio Data* are calculated as:

$$\text{ExpectedAudioData} = R_i \times \text{BaseRTT} \quad (5.4)$$

$$\text{ActualAudioData} = R_i \times \text{RTT}_i \quad (5.5)$$

where R_i (kbps) is the audio codec bit rate used for voice flow i . RTT_i is the round-trip time between the gateway and the wireless station i estimated when the receiver report i is transmitted through the gateway. *Actual Audio Data* represents the amount of audio data transmitted during RTT_i using codec rate R_i . The difference *Diff* is calculated as:

$$\begin{aligned} \text{Diff} &= (\text{ActualAudioData} - \text{ExpectedAudioData}) / \text{RTT}_i \\ &= R_i \left(1 - \frac{\text{BaseRTT}}{\text{RTT}_i}\right) \end{aligned} \quad (5.6)$$

Diff is an estimation of the extra audio data rate related to the voice flow i , i.e., audio data that would not have been sent during RTT_i if the audio codec used for this voice flow exactly matched the available wireless channel bandwidth. Since $\text{BaseRTT} \leq \text{RTT}_i$, *Diff* is typically positive.

The algorithm then compares the value of *Diff* to α and β thresholds, these two thresholds are defined in terms of data rate (kbps). The farther away the actual audio rate gets from the expected value, the more congestion there is in the WLAN, which implies that the sending codec rate should be reduced. This decrease is triggered by the β threshold. The α threshold triggers the increase of the audio codec rate in case the voice flow is not utilizing the available bandwidth. The goal is then to keep between α and β the difference between the expected and the actual codec rate of voice flows transmitted over the wireless channel (Flow-chart described in Figure 5.3).

In the following, we describe the proposed congestion and wireless loss differentiation scheme. The main idea of this scheme is that the loss differentiation mechanism would predict that next packet loss will be due to congestion when the Vegas-like audio rate control algorithm (described above) suggests to decrease audio codec rate (after detecting an RTT increase). If a loss occurs when the algorithm is recommending increasing codec rate (after detecting an RTT decrease), it may be reasonable to assume that the loss is due to transmission errors on the wireless channel and thus adding FEC information is recommended in order to resolve this loss (Figure 5.3). If the codec rate was not varied, the next packet loss will also be considered as a wireless loss.

In fact, queuing delays decrease when the service rate increases or when the traffic load decreases, i.e., when network bandwidth increases. A queue build-up typically results in congestion losses. However, wireless transmission losses occur independently of the network conditions [114]. Therefore, in our loss prediction scheme, a lack of an indication of a congestion loss ($\text{Diff} > \beta$) is

used as a wireless loss indication.

The mathematical explanation for this loss prediction mechanism is the following. Let us express RTT_i as $RTT_i = \text{BaseRTT} + dq_i$ where dq_i is the extra queuing delay for the flow i , as compared to BaseRTT (assuming that the round trip time variation is due only to the queuing delays). Thus,

$$\text{Diff} = R_i \left(1 - \frac{\text{BaseRTT}}{\text{BaseRTT} + dq_i}\right) \quad (5.7)$$

Then we obtain:

$$\frac{\delta(\text{Diff})}{\delta dq_i} = \frac{R_i \text{BaseRTT}}{(\text{BaseRTT} + dq_i)^2} \quad (5.8)$$

Since $\frac{\delta(\text{Diff})}{\delta dq_i} > 0$, the rate difference Diff increases with increasing queuing delay dq_i , which represents a congestion indication.

5.5 Simulation Experiments and Discussion

We provide NS-2 [37] simulation results obtained from downlink VoIP flows transmitted on a WLAN with CBR background traffic using the EDCF/HCF mode of operation. We consider that EDCF and HCF have the same duration of 50% of the beacon period. We consider a high-rate IEEE 802.11a WLAN with physical data rate of 36 Mbps and an adaptive system in which sources can switch between five bit-rates, corresponding to widely used telephone voice coding standards (Table 5.1).

Table 5.1: Codec bit rate and packet size

Codec	Bit Rate (Kbit/s)	Payload Size (bytes)	Total Packet Size (with IP/UDP/RTP headers)
G.711	64	160	200
G.726	40	100	140
G.726	32	80	120
G.726	24	60	100
G.726	16	40	80

The G.726 [20] codec makes a conversion of a 64 kbit/s pulse code modulation (PCM) channel to and from a 40, 32, 24 or 16 kbit/s channel. The conversion is applied to the PCM bit stream using an ADPCM (Adaptive Differential Pulse Code Modulation) transcoding technique. The relationship between the voice frequency signals and the PCM encoding/decoding laws is fully specified in Recommendation G.711 [17]. A variable bit-rate system operating at such bit-rates can be viewed as a system that always delivers “toll quality,” but with different levels of complexity, delay and robustness.

Codec rates and packet sizes used to simulate our mechanism are shown in Table 5.1. For TFRC flows, we use packets of 200 bytes and for background flows we set packet size to 1500 bytes. The

goal of the simulations is not a complete analysis of the considered system but, rather, an indication that interesting performance evaluation indices can be derived through the proposed approach. We consider only the adaptive codec rate part of the Vegas-like control algorithm. In future work we will consider adding FEC information based on the loss prediction mechanism. Simulations are carried out for the duration of 20 seconds and the presented results are averaged over 5 simulations. We set the EDCF VoIP flow priority to 6 and background flows priority to 1. The system parameters used for these simulations are listed in Table 5.2.

Table 5.2: Simulation system parameters

Parameter	Value
Beacon period	100 ms
Slot time (T_{SLOT})	9 μs
SIFS (T_{SIFS})	10 μs
CWmin_6 (Voice)	7
CWmax_6 (Voice)	15
CWmin_1 (BG)	31
CWmax_1 (BG)	1023

In these simulations, the number of stations is increased from 2 to 24 (including the QAP). A Vegas-like VoIP flow and a background flow are transmitted from the QAP to each QSTA. We use $\alpha = 10$ kbps and $\beta = 20$ kbps parameters for the Vegas-like audio rate control algorithm.

Figure 5.4 shows the average delay (queuing and transmission delay) of VoIP traffic over the WLAN. Adaptive audio rate control presents good performance, as it keeps average delay below 4 ms in both situations of small and large number of VoIP flows.

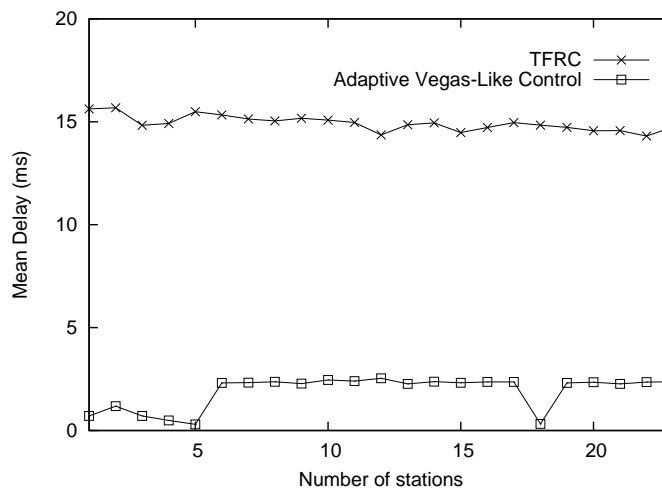


Figure 5.4: Mean VoIP delay vs. the number of QSTAs

With TFRC, VoIP average delay rises above 15 ms. The confirmation of these results is provided

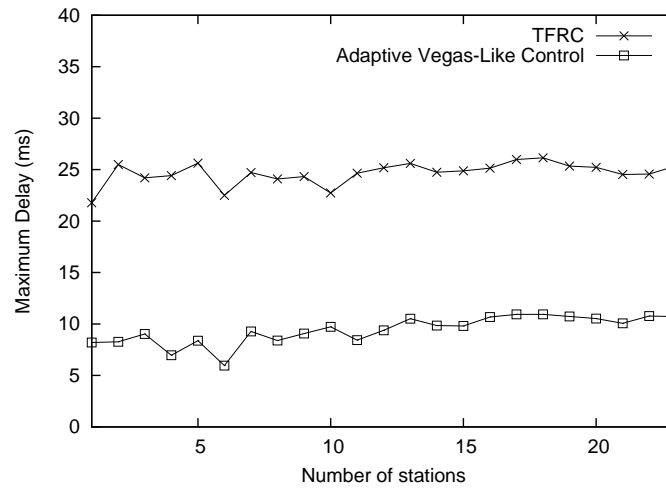


Figure 5.5: Maximum VoIP delay vs. the number of QSTAs

by the maximum voice packet delay depicted in Figure 5.5. Adaptive audio rate control keeps the maximum delay below 10 ms. For TFRC the maximum delay is about 25 ms. This delay reduction can be explained by the codec bit rate decrease in response to congestion at the QAP queue (indicated by the increase of RTT value). Reducing the audio packet delay by the value of about 10 ms on the WLAN is important in order to cope with the before mentioned audio QoS requirements (one way delay is restricted to at most 150 ms) and since we have to consider the delay caused in the wide area network that must be traversed by an audio packet on its way to the destination in the WLAN. The adaptability of our control mechanism to the WLAN conditions ensures a reduced packet voice delay and this improves perceived voice quality. Figure 5.6 depicts the mean bandwidth of VoIP flows as the number of stations is increased. When the number of flows is below 8, the mean bandwidth of our adaptive VoIP mechanism is higher than that of TFRC.

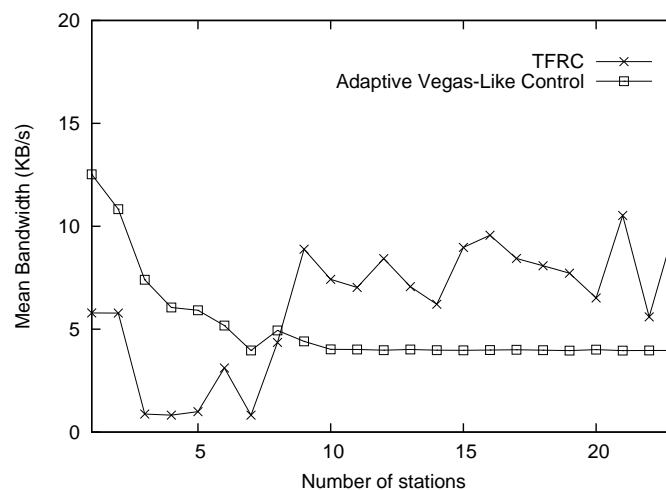


Figure 5.6: Mean bandwidth of VoIP flows vs. the number of QSTAs

Our mechanism is less conservative than TFRC when the number of flows is reduced, and this avoids the under-utilization of network bandwidth and also avoids the voice transmission quality degradation in response to congestion. For more than 6 flows, the mean bandwidth used by our mechanism is steady (4 KB/s) and smoother than TFRC mean bandwidth. Maintaining low sending rate variation and avoiding abrupt rate changes will reduce the delay jitter and this will ameliorate the perceived voice quality. Besides, reducing the sending rate in case of high load network conditions will increase the WLAN capacity (as shown in Chapter 4).

In order to evaluate the fairness between VoIP flows, we compute for each scheme the fairness index defined as:

$$\text{FairnessIndex} = \frac{(\sum_{i=1}^n T_i)^2}{n \times \sum_{i=1}^n (T_i)^2} \quad (5.9)$$

where n is the number of flows using the same control scheme and T_i is the throughput of flow i .

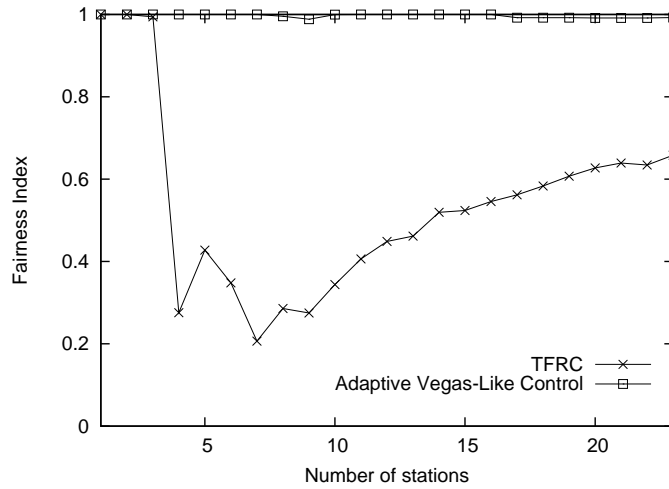


Figure 5.7: Improved VoIP flow fairness with the adaptive Vegas-like codec rate control

The fairness index is equal to 1 if all T_i are equal (highest degree of fairness between flows). Figure 5.7 shows that our adaptive audio rate control achieves considerably better fairness than TFRC. With our mechanism, fairness index is kept almost at 1 independently of the number of VoIP flows, however with TFRC, this index goes below 0.4 for a small number of VoIP flows (6 flows) and it is improved when the number of flows is increased (0.62 for 20 flows). This can be explained by the fact that our adaptive control uses delay information to avoid congestion and adapts the audio rate of each flow, however TFRC uses loss information for rate adaptation and this information is not so accurate for detecting congestion in WLANs.

5.6 Conclusions

In this chapter, we have proposed a novel adaptive architecture for the transport of VoIP traffic over heterogeneous wired/wireless Internet environments. This architecture supports adaptive VoIP coding on WLAN using a VoIP gateway located at the edge of the wired Internet and the wireless last-hop link. The adaptive coding mechanism uses a delay-based congestion avoidance technique that allows the differentiation between wireless and congestion losses. We illustrated the proposed scheme considering a specific control mechanism applied to variable bit-rate system operating at five VoIP coding bit rates (64, 40, 32, 24 and 16 Kbit/s) . Simulation results show that our adaptive architecture responds constructively to network congestion and improves QoS support for VoIP in IEEE 802.11e networks. Using the 802.11e EDCF/HCF scheme, we reduce the transmission delay of VoIP traffic compared to current TFRC algorithm. Obtained results also show that our adaptive rate control mechanism is fairer than TFRC especially when the number of VoIP flows is increased. The system capacity is also increased, since the sending rate is reduced in case of high-load network conditions.

Chapter 6

Conclusions and Perspectives

In this thesis, we have studied large scale Voice over IP deployment over heterogeneous Internet environments. We focused on the case of large number of VoIP communications transmitted simultaneously over network links. We have addressed problems related to three main research topics: (i) congestion control mechanisms for unresponsive VoIP traffic, (ii) capacity evaluation of VoIP in IEEE 802.11e WLANs and (iii) adaptive VoIP transmission mechanisms in heterogeneous Internet environments with wireless last-hop access link. Ideas presented in this dissertation provide a significant step towards an efficient and scalable VoIP deployment that improves VoIP transmission performance. We summarize in the following the research contributions of this work.

In Chapter 2, we presented an overview of VoIP system including basic mechanisms of voice media coding and transport over the Internet, features of standard VoIP codecs (i.e., G.711, G.726, G.729, G.723.1), protocols used for the transport of real-time voice traffic over the Internet (i.e., RTP/RTCP), and VoIP quality impairments (low bit rate codecs, delay, loss, jitter). The issues we addressed in this chapter are:

- The tradeoff between bandwidth efficiency and VoIP transmission quality mainly affected by codec bit rate, delay and packet loss rate.
- The effect of network characteristics (i.e., bandwidth, infrastructure heterogeneity) and congestion state on the performance of VoIP transmission.

In Chapter 3, we studied the need to design congestion control for the growing class of VoIP traffic and we introduced a VoIP congestion control scheme called Voice-TFCC (Voice TCP-Friendly Congestion Control). Voice-TFCC combines RTP voice flow multiplexing and TCP-friendly congestion control mechanism in order to dynamically adapt packet and codec rate of VoIP flows while being fair with coexisting Internet traffic sharing the same network links (i.e., TCP traffic). We evaluate the performance of our scheme, and demonstrate its feasibility using a prototype implementation. Experimental results conducted over PlanetLab testbed network highlighted the benefits of adaptive packet rate and codec rate on VoIP transmission performance. Indeed, significant delay improvements were obtained especially in the case of bandwidth limited network environments. In addition, flow multiplexing reduces the protocol header overhead and codec rate adaptation improves

the bandwidth efficiency. Applying TCP-friendly congestion control mechanism ensures fairness with Internet traffic sharing network links with VoIP traffic. We believe that the use of congestion control mechanisms for VoIP flows represents a promising solution to prevent the degradation of end-to-end congestion control in the Internet, by providing a viable alternative behavior for these flows. Future works related to this part of our contributions can involve many issues. Here we point some of them.

- Study the effect of Voice-TFCC flow multiplexing in the case of VoIP sources implementing silence suppression mechanism. VBR source model should be considered in that case.
- Evaluate various techniques for the loss rate estimation and their effect on TCP-friendly mechanism accuracy.
- A promising extension to the Voice-TFCC scheme is to study the case of high traffic load caused by a large number of VoIP flows occurring at the source gateway and that can not be multiplexed within a TCP-friendly flow. In that case, one solution is to forward some of the flows through a third gateway (between source and destination) in order to reduce traffic load and to achieve the TCP-friendliness condition. For the gateway selection, peer-to-peer algorithms can be used to find the best Internet path between peer VoIP gateways in terms of path quality mainly estimated by delay and bandwidth criterions. Another solution is to consider path switching mechanism where VoIP flows are redirected towards a gateway presenting network path conditions that are better than those obtained through the current gateway. Signaling protocols such as SIP could be used for the communication between VoIP gateways and terminals.

In Chapter 4, we focused on the capacity evaluation of IEEE 802.11e networks in terms of interactive VoIP communications. We have introduced an analytical VoWLAN capacity model and we have presented numerical results derived from its application on particular WLAN configurations. The capacity of VoIP over 802.11e networks is reduced mainly because of:

- Packet-header overhead of the standard protocol stack (IP/UDP/RTP/MAC/PHY) added to short voice packets.
- The inefficiency inherent in the WLAN medium-access control (MAC) protocol (i.e., IFS time intervals, backoff mechanisms, ACK mechanisms).

We studied the case in which QSTAs are allowed to transmit voice frames once per superframe during a TXOP obtained by a successful EDCA contention (EDCA-TXOP) or by the reception of a QoS poll frame from the QAP (HCCA-TXOP). Our modeling results show that using both EDCA and HCCA modes of operation to transmit voice significantly increases the maximum number of supported VoIP communications. We also show that CAPs (Controlled Access Periods) allow higher VoIP capacity than CPs (Contention Periods). During CPs transmission efficiency drops because of contention mechanism. Our results also showed that compared to G.711 codec, the use of low bit

rate codecs (i.e., G.729, G.723.1) significantly improves the VoIP capacity in 802.11e networks. We showed that optimal VoWLAN quality can be obtained by the use of codec adaptation mechanism that determines the voice codec to be used according to the number of voice calls being transmitted over the WLAN.

Future work on this contribution will first involve proposing a VoIP capacity model that considers the effect of data traffic (i.e., video, best-effort traffic), transmitted over the 802.11 channel simultaneously with voice traffic, on the performance of supported VoIP communications (i.e., packet delay and loss). In particular, the effect of backoff time increase caused by frame collisions should be studied.

Another interesting extension to this work is to design a joint capacity/quality optimization scheme in which a VoIP gateway interacts with the QAP. The QAP adapts MAC parameters which determine the number of admissible VoIP calls (i.e., the superframe (SF) size and the service interval (SI)), then the voice codec that ensures optimal quality is determined according to the number of accepted voice calls.

Chapter 5 concerned adaptive VoIP transmission over heterogeneous wired/wireless networks. We believe that the heterogeneity growth of current Internet environments will be the main challenge for networking in the coming years. Research perspectives on Internet applications, in particular VoIP, should take this into consideration by focusing on the design of adaptive protocols that dynamically adjust their behavior according to the network congestion state and by the use of intermediate nodes to efficiently interconnect heterogeneous networks.

We proposed a network architecture that is based on a VoIP gateway located at the edge of WLAN and associated with an 802.11e QoS enhanced access point (QAP). The gateway is used to transcode voice flows before their transmissions over the wireless channel. The instantaneous bit rate is determined by a control mechanism based on the estimation of the wireless channel congestion state. Our mechanism dynamically adapts audio codec bit rate using a delay-based congestion avoidance technique so as to preserve acceptable levels of voice quality and to ensure fairness with TCP traffic. The proposed adaptation mechanism also allows the differentiation between congestion and wireless losses in order to respond appropriately (reduce coding rate/add FEC redundant information). Simulation results show that the proposed mechanism improves voice transmission performance, especially in the case of large number of VoIP flows.

Future work related to this contribution includes performing simulations of the proposed loss prediction mechanism in order to provide additional insight into packet losses due to congestion and wireless errors. Another important future work is to enhance the adaptive rate/loss control mechanism based on the WLAN capacity model proposed in Chapter 4 in order to optimally select voice/FEC coding rate and 802.11e MAC parameters that improve VoIP transmission quality as well as WLAN capacity. The use of RTS/CTS option for MAC frame error protection can be addressed. A dynamic decision to enable RTS/CTS option should be taken based on the voice frame size that varies according to the service interval (determined by the QAP) and the voice coding rate (determined by the VoIP application). Given that small voice frames encounter less collisions than larger ones and that RTS/CTS overhead is higher for small frames, RTS/CTS option should be enabled

only to protect large voice frames against wireless channel errors.

Besides, it would be interesting to define a standardized communication interface between layers (i.e., MAC and application layer) to allow better loss and rate control mechanisms for VoIP in WLAN environments. Current wireless standards do not propose such interface. With the next generation of WLAN standards on the horizon, especially 802.11n, such cross layer interaction should be considered in order to improve VoIP applications performance while preserving advantages of layered architecture.

Bibliography

- [1] D. Cohen, *Issues in transnet packetized voice communications*, in Proc. of Fifth Data Communications Symposium, Snowbird, Utah, pp. 6-10, ACM, IEEE, September 1977.
- [2] C. N. Chuah, *A scalable framework for IP-network resource provisioning through aggregation and hierarchical control*, PhD thesis, University of California at Berkeley, 2001.
- [3] ITU-T Recommendation G.114, *One way transmission time*, International Telecommunication Union, May 2003.
- [4] Skype Technologies S. A., URL <http://www.skype.org>, 2002-2006.
- [5] R. Braden, D. Clark, and S. Shenker, *Integrated Services in the Internet architecture: an overview*, RFC 1633, June 1994.
- [6] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, *Resource ReSerVation Protocol (RSVP), Version 1 Functional Specification*, RFC 2205, September 1997.
- [7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, *An architecture for Differentiated Services*, RFC 2475, December 1998.
- [8] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, *RTP: A transport protocol for real-time applications*, RFC 3550, July 2003.
- [9] H. Schulzrinne and S. Casner, *RTP profile for audio and video conferences with minimal control*, RFC 3551, July 2003.
- [10] R. Ramachandran, J. Kurose, D. Towsley, and H. Schulzrinne, *Adaptive playout mechanisms for packetized audio applications in wide-area networks*, in Proc. of INFOCOM'94, Toronto, Canada, June 1994.
- [11] J-C. Bolot, S. Fosse-Parisis, and D. Towsley, *Adaptive FEC-based error control for Internet telephony*, in Proc. of INFOCOM'99, New York, March 1999.
- [12] C. Boutremans and J. Y. Le Boudec, *Adaptive joint playout buffer and FEC adjustment for internet telephony*, in Proc. of IEEE INFOCOM'03, San-Francisco, CA, April 2003.
- [13] J. D. Padhye, *Towards a comprehensive congestion control framework for continuous media flows in best effort networks*, PhD thesis, University of Massachusetts Amherst, 2000.

- [14] J. E. Cohen and F. P. Kelly, *A paradox of congestion in a queueing network*, Journal of Applied Probability, vol. 27, pp. 730–734, 1990.
- [15] J. Woodward, *Speech coding*, Communications Research Group, University of Southampton, URL http://www-mobile.ecs.soton.ac.uk/speech_codecs/, August 2005.
- [16] O. Hersent, D. Gurle, and J-P. Petit, *IP Telephony: Packet-based multimedia communications systems*, Addison-Wesley, Harlow, England, 2000.
- [17] ITU-T Recommendation G.711, *Pulse Code Modulation (PCM) of voice frequencies*, International Telecommunication Union, November 1988.
- [18] ITU-T Recommendation G.729, *Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)*, International Telecommunication Union, March 1996.
- [19] ITU-T Recommendation G.723.1, *Speech coders : Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*, International Telecommunication Union, March 1996.
- [20] ITU-T Recommendation G.726, *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)*, International Telecommunication Union, December 1990.
- [21] Global IP Sound (GIPS), *Speech coding and speech quality in IP Telephony*, Stockholm, Sweden, October 2001.
- [22] ITU-T Recommendation G.728, *Coding of speech at 16 kbit/s using low-delay code excited linear prediction*, International Telecommunication Union, September 1992.
- [23] ITU-T Recommendation G.723.1 Annex A, *Speech coders : Silence compression scheme*, International Telecommunication Union, March 1996.
- [24] ITU-T Recommendation G.729 Annex A, *Reduced complexity 8 kbit/s CS-ACELP speech codec*, International Telecommunication Union, November 1996.
- [25] ITU-T Recommendation G.729 Annex B, *A silence compression scheme for G.729 optimized for terminals conforming to Recommendation V.70*, International Telecommunication Union, October 1996.
- [26] European Telecommunications Standards Institute (ETSI), *Digital cellular telecommunications system full rate speech transcoding*, GSM 6.10 version 5.1.1, May 1998.
- [27] J-F. Susbielle, *Internet multimédia et temps réel*, éditions Eyrolles, Paris, France, 2000.
- [28] J. Postel, *Internet Protocol*, RFC 760, USC/Information Sciences Institute (ISI), January 1980.
- [29] J. Postel, *User Datagram Protocol*, RFC 768, Information Sciences Institute (ISI), August 1980.

- [30] H. Schulzrinne and J. Rosenberg, *Internet Telephony: architecture and protocols,, an IETF perspective*, Computer Networks, special issue on Internet Telephony, Volume 31, Issue 3, February 1999.
- [31] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, *SIP: Session Initiation Protocol*, RFC 3261, June 2002.
- [32] M. Handley and V. Jacobson, *SDP: Session Description Protocol*, RFC 2327, April 1998.
- [33] ITU-T Recommendation H.323, *Packet-based multimedia communications systems*, International Telecommunication Union, November 2000.
- [34] A. Trad and H. Afifi, *TFMC: a TCP-Friendly Multiplexing Control Scheme for VoIP Flow Transmission*, in Proc. of IEEE ICN'04, Guadeloupe, France, March 2004.
- [35] A. Trad, Q. Ni, and H. Afifi, *Adaptive VoIP transmission over heterogeneous wired/wireless networks*, in Proc. of MIPS'04, Grenoble, France, November 2004.
- [36] A. Trad, F. Munir, and H. Afifi, *Capacity Evaluation of VoIP in IEEE 802.11e Network Environment*, in Proc. of IEEE CCNC'06, Las Vegas, Nevada, USA, January 2006.
- [37] *The Network Simulator NS-2*, URL: <http://www.isi.edu/nsnam/ns>.
- [38] J. Rosenberg, *Distributed algorithms and protocols for scalable Internet telephony*, PhD thesis, Columbia University, 2001.
- [39] D. L. Mills, *Network Time Protocol specification, implementation and analysis*, RFC 1305, University of Delaware, March 1992.
- [40] J.-C. Bolot, *End-to-end packet delay and loss behavior in the Internet*, in Proc. of SIGCOM'93, N. Y., USA, September 1993.
- [41] J.-C. Bolot, H. Crepin, and A. V. Garcia, *Analysis of audio packet loss in the Internet*, in Proc. of NOSSDAV'95, New Hampshire, USA, April 1995.
- [42] A. P. Markopoulou, F. A. Tobagi, and M. J. karam, *Assessment of VoIP quality over Internet backbones*, in Proc. of IEEE INFOCOM'02, New York, USA, June 2002.
- [43] S. Shenker, C. Partridge, and R. Guerin, *Specication of guaranteed Quality of Service*, RFC 2212, September 1997.
- [44] J. Wroclawski, *Specication of the controlled-load network element service*, RFC 2211, September 1997.
- [45] L. Mathy, C. Edwards, and D. Hutchison, *The Internet: a global telecommunications solution?*, IEEE Network Magazine, Vol. 4657, April 2000.

- [46] H. Sanneck and N. T. L. Le, *Speech property-based FEC for Internet telephony applications*, in Proc. of the SPIE/ACM SIGMM MMCN conference, San Jose, California, January 2000.
- [47] N. S. Jayant, *Effects of packet losses on waveform-codec speech*, in Proc. of IEEE International Conference on Computer Communications, Atlanta, Georgia, October 1980.
- [48] ITU-T Recommendation P.862, *Perceptual evaluation of speech quality (PESQ): an objective method of end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*, International Telecommunication Union, January 2001.
- [49] ITU-T Recommendation G.107, *The Emodel, a computational model for use in transmission planning*, International Telecommunication Union, March 2005.
- [50] ITU-T Recommendation G.108, *Application of the Emodel: a planning guide*, International Telecommunication Union, September 1998.
- [51] ITU-T Recommendation G.109, *Definition of categories of speech transmission quality*, International Telecommunication Union, September 1999.
- [52] ITU-T Recommendation G.113, *Transmission impairments due to speech processing*, International Telecommunication Union, February 2001.
- [53] ITU-T Recommendation G.113, *Appendix I: Provisional planning values for the equipment impairment factor I_e and packet-loss robustness factor B_{pl}* , International Telecommunication Union, May 2002.
- [54] R. G. Cole and J. H. Rosenbluth, *Voice over IP performance monitoring*, Computer Communication Review, vol. 31, no. 2, pp. 924, April 2001.
- [55] V. Jacobson and M. Karels, *Congestion avoidance and control*, ACM Computer Communication Review, 18(4):314-329, August 1990.
- [56] ITU-T Recommendation P. 59, *Artificial conversations speech*, International Telecommunication Union, March 1993.
- [57] C. Groves, M. Pantaleo, T. Anderson, T. Taylor, *Gateway Control Protocol Version 1*, RFC 3525, June 2003.
- [58] Cisco Systems, *Understanding packet voice protocols*, white paper, 2001.
- [59] P. Brady, *A model for generating On-Off speech patterns in two-way conversation*, Bell Syst. Tech. Journal, vol. 48, no. 7, pp. 2245-2272, September 1969.
- [60] C. Perkins, J. Rosenberg and H. Schulzrinne, *RTP Testing Strategies*, RFC 3158, August 2001.
- [61] Y. R. Yang and S. Lam, *General AIMD congestion control*, Technical Report TR-2000-09, Department of Computer Sciences, University of Texas, May 2000.

- [62] R. Johnari and D. K. H. Tan, *End-to-end congestion control for the Internet: delays and stability*, IEEE/ACM Transactions on Networking, December 2001.
- [63] S. Floyd and J. Kempf, *IAB Concerns Regarding Congestion Control for Voice Traffic in the Internet*, RFC 3714, March 2004.
- [64] S. Floyd and V. Jacobson, *Random Early Detection gateways for congestion avoidance*, IEEE/ACM Transactions on Networking, August 1993.
- [65] J. Ash, B. Goode, J. Hand, and R. Zhang, *Requirements for end-to-end VoIP header compression*, Internet Draft, Network Working Group, February 2003.
- [66] S. Casner and V. Jacobson, *Compressing IP/UDP/RTP headers for low speed serial links*, RFC 2508, February 1999.
- [67] C. Bormann, C. Burmeister, M. Degermark, H. Fukushima, H. Hannu, L-E. Jonsson, R. Hakenberg, T. Koren, K. Le, Z. Liu, A. Martensson, A. Miyazaki, K. Svanbro, T. Wiebke, T. Yoshimura, and H. Zheng, *RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed*, RFC 3095, July 2001.
- [68] H. P. Sze, S. C. Liew, J. Y. B. Lee, and C. S. Yip, *A multiplexing scheme for H.323 Voice-Over-IP Applications*, IEEE Journal on Selected Areas in Communications, Vol. 20, No. 7, September 2002.
- [69] T. Hoshi, K. Tanigawa, and T. Takahashi, *IP telephony gateway and Its media stream control*, Hitachi Review, Vol. 48, No. 4, 1999.
- [70] K. Tanigawa, T. Hoshi, and K. Tsukada, *Simple RTP multiplexing transfer methods for VoIP*, Internet Draft draft-tanigawa-rtp-multiplex-01.txt, May 1999.
- [71] R. Pazhyannur, I. Ali, and C. Fox, *PPP multiplexing*, RFC 3153, August 2001.
- [72] S. Floyd and E. Kohler, *TCP Friendly Rate Control (TFRC) for Voice: VoIP Variant*, Internet Draft, July 2005.
- [73] B. Thompson, T. Koren, and D. Wing, *Tunneling Multiplexed Compressed RTP (TCRTP)*, RFC 4170, November 2005.
- [74] J. Rosenberg and H. Schulzrinne, *Issues and options for an aggregation service within RTP*, Internet Draft, IETF Audio/Video Transport Working Group, May 1997.
- [75] J. Rosenberg and H. Schulzrinne, *An RTP payload format for user multiplexing*, Internet Draft, Audio/Video Transport Working Group, November 1998.
- [76] K. El-Khatib, G. Luo, G. Bochmann, and P. Feng, *Multiplexing scheme for RTP flows between access routers*, Internet Draft, Audio/Video Transport Working Group, April 2000.

- [77] *PlanetLab: An open platform for developing, deploying, and accessing planetary-scale services*, URL <http://www.planet-lab.org>, 2004-2005 PlanetLab Consortium.
- [78] J. Mahdavi and S. Floyd, *TCP-Friendly unicast rate-based flow control*, note sent to end2end-interest mailing list, January 1997.
- [79] S. Floyd and K. Fall, *Promoting the use of end-to-end congestion control in the Internet*, IEEE/ACM Transactions on Networking, August 1999.
- [80] T. Turletti, S. Parisi, and J. Bolot, *Experiments with a layered transmission scheme over the Internet*, Technical Report RR-3296, INRIA, France, November 1997.
- [81] D. Sisalem and H. Schulzrinne, *The Loss-Delay based Adjustment algorithm: a TCP-friendly adaptation scheme*, in Proc. of NOSSDAV'98, pp. 215-226, July 1998.
- [82] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, *Modeling TCP throughput: a simple model and its empirical validation*, in Proc. of ACM SIGCOMM'98, Vancouver, B.C., Canada, September 1998.
- [83] M. Handley, S. Floyd, J. Padhye, and J. Widmer, *TCP Friendly Rate Control (TFRC): protocol specification*, RFC 3448, January 2003.
- [84] S. Floyd, M. Handley, J. Padhye, and J. Widmer, *Equation-based congestion control for unicast applications*, in Proc. of ACM SIGCOMM'2000, pp. 43-56, Stockholm, Sweden, August/September 2000.
- [85] IEEE Std 802.11-1999, Part 11: *Wireless LAN MAC and physical layer specifications*, Reference number ISO/IEC 8802-11:1999(E).
- [86] IEEE 802.11e/D9.0, *Wireless MAC and physical layer specifications: Medium Access Control (MAC) Quality of Service (QoS) enhancements*, August 2004.
- [87] IEEE 802.11b, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Higher speed physical layer extension in the 2.4 GHz band*, IEEE 802.11b WG part 11b, 1999.
- [88] IEEE 802.11a, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, High-speed physical layer in the 5 GHz band*, IEEE 802.11a WG part 11a, 1999.
- [89] IEEE 802.11g, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Further higher speed physical layer extension in the 2.4 GHz band*, IEEE 802.11a WG part 11g, 2003.
- [90] Q. Ni, *Performance analysis and enhancements for IEEE 802.11e wireless networks*, IEEE Network, July/August 2005.

- [91] S. Garg and M. Kappes, *An experimental study of throughput for UDP and VoIP traffic in IEEE 802.11b networks*, In Proc. of the IEEE WCNC'03, New Orleans, LA, 2003.
- [92] M. Veeraraghavan, N. Cocker, and T. Moors, *Support of voice services in IEEE 802.11 wireless LANs*, in Proc. of INFOCOM'01, Alaska, April 2001.
- [93] E. Ziouva and T. Antonakopoulos, *Efficient voice communications over IEEE 802.11 WLANs using improved PCF procedures*, in International Network Conference, INC, 2002.
- [94] D.-Y. Chen, S. Garg, M. Kappes, and K. S. Trivedi, *Supporting VBR VoIP traffic with IEEE 802.11 WLAN in PCF mode*, in Proc. of OPNETWork 2002, Washington D. C., August 2002.
- [95] S. Garg and M. Kappes, *Can I add a VoIP call?*, in Proc. of IEEE ICC'03, Alaska, May 2003.
- [96] N. Snavatkul, Y. Chen, and S. Emeott, *Voice capacity evaluation of IEEE 802.11a with automatic rate selection*, in Proc. of GLOBECOM'03, San Francisco, December 2003.
- [97] D. P. Hole and F. A. Tobagi, *Capacity of IEEE 802.11b Wireless LAN supporting VoIP*, in Proc. of IEEE ICC'04, Paris, June 2004.
- [98] T. Kawata, S. Shin, A. G. Forte, and H. Schulzrinne, *Using dynamic PCF to improve the capacity for VoIP traffic in IEEE 802.11 networks*, in Proc. of IEEE WCNC'05, New Orleans, USA, March 2005.
- [99] W. Wang, S. C. Liew, and V. O. K. Li, *Solutions to performance problems in VoIP over a 802.11 wireless LAN*, IEEE transaction on Vehicular Technology, vol. 54, January 2005.
- [100] K. Medepalli, P. Gopalakrishnan, D. Famolari, and T. Kodama, *Voice capacity of IEEE 802.11b, 802.11a and 802.11g Wireless LANs*, in Proc. of IEEE GLOBECOM'04, Dallas, Texas, November 2004.
- [101] M. Chen and A. Zakhori, *Rate control for streaming video over wireless*, in Proc. of INFOCOM'04, Hong Kong, March 2004.
- [102] J. Matta, C. Pépin, K. Lashkari, and R. Jain, *A source and channel rate adaptation algorithm for AMR in VoIP using the Emodel*, in Proc. of NOSSDAV'03, San-Francisco, California, June 2003.
- [103] R. Rejaie, M. Handley, and D. Estrin, *RAP: an end-to-end rate-based congestion control mechanism for realtime streams in the Internet*, in Proc. of IEEE INFOCOM'99, New York, March 1999.
- [104] D. Sisalem and H. Schulzrinne, *The Loss-Delay based Adjustment algorithm: a TCP-friendly adaptation scheme*, in Proc. of NOSSDAV'98, Cambridge, UK, July 1998.
- [105] I. Rhee, V. Ozdemir, and Y. Fi, *TEAR: TCP Emulation at Receivers – flow control for multimedia streaming*, NCSU Technical Report, April 2000.

- [106] A. Grilo and M. Nunes, *Performance evaluation of 802.11e*, in Proc. of IEEE PIMRC'02, Lisboa, Portugal, September 2002.
- [107] M. Karam and F. Tobagi, *Analysis of the delay and jitter of voice traffic over the Internet*, in Proc. of IEEE INFOCOM'01, Anchorage, AL, April 2001.
- [108] H. Balakrishnan, S. Seshan, and R. H. Katz, *Improving reliable transport and handoff performance in cellular wireless networks*, ACM Wireless Networks, vol. 1, no. 4, 1995.
- [109] K. Ramakrishnan, S. Floyd, and D. Black, *The addition of Explicit Congestion Notification (ECN) to IP*, RFC 3168, September 2001.
- [110] S. Biaz and N. H. Vaidya, *Discriminating congestion losses from wireless losses using inter-arrival times at the receiver*, in IEEE Symposium ASSET'99, Texas, USA, March 1999.
- [111] Y. Tobe, Y. Tamura, A. Molano, S. Ghosh, and H. Tokuda, *Achieving moderate fairness for UDP flows by pathstatus classification*, in Proc. of IEEE CLCN'00, Tampa, FL, November 2000.
- [112] L. Brakmo and L. L. Peterson, *TCP Vegas: End to end congestion avoidance on a global Internet*, IEEE Journal of Selected Areas in Communication, 13:1465-1480, 1995.
- [113] L. Brakmo and S. O'Malley, *TCP-Vegas: New techniques for congestion detection and avoidance*, In Proc. of ACM SIGCOMM'94, pp. 24-35, October 1994.
- [114] S. Biaz and N. H. Vaidya, *Distinguishing congestion losses from wireless transmission losses: A negative result*, In Proc. of IEEE IC3N, New Orleans, USA, October 1998.
- [115] S. Cen, P. C. Cosman, and G. M. Voelker, *End-to-end differentiation of congestion and wireless losses*, in Proc. of MMCN'02, San Jose, CA, January 2002.

Acronyms Glossary

ABS	Analysis-By-Synthesis
AC	Access Category
AIFS	Arbitration Inter Frame Space
AIMD	Additive Increase Multiplicative Decrease
AP	Access Point
CA	Collision Avoidance
CAP	Controlled Access Period
CBR	Constant Bit Rate
CC	Congestion Control
CELP	Code-Excited Linear Prediction
CFP	Contention Free Period
CNG	Comfort Noise Generation
CP	Contention Period
cRTP	compressed RTP
CSMA	Carrier-Sense Multiple Access
CTS	Clear-To-Send
CW	Contention Window
DiffServ	Differentiated Services
DIFS	Distributed Inter Frame Space
DPCM	Differential Pulse Code Modulation
DTX	Discontinuous Transmission

EDCA	Enhanced Distributed Coordinated Access
EDCF	Enhanced Distributed Channel Function
EIFS	Extended Inter Frame Space
ETSI	European Telecommunications Standards Institute
FEC	Forward Error Correction
FIFO	First In First Out
GSM	Global System for Mobile communications
HC	Hybrid Coordinator
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IntServ	Integrated Services
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ITU	International Telecommunication Union
LPC	Linear Predictive Coding
MIPS	Million Instructions Per Second
MOS	Mean Opinion Score
MTU	Maximum Transfer Unit
PCM	Pulse Code Modulation
PESQ	Perceptual Evaluation of Speech Quality
PIFS	Point coordination Inter Frame Space
PLC	Packet Loss Concealment
PSTN	Public Switched Telephone Network (equivalent to RTC)
QAP	QoS-enhanced Access Point

QoS	Quality of Service
QSTA	QoS STation
RED	Random Early Detection
RFC	Request For Comments
RPE	Regular-Pulse Excited
RSVP	Resource reSerVation Protocol
RTCP	Real-time Transport Control Protocol
RTP	Real-time Transport Protocol
RTS	Request-To-Send
RTT	Round-Trip Time
RTO	Retransmit Time Out
SDP	Session Description Protocol
SF	Super Frame
SIFS	Short Inter Frame Space
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
TFRC	TCP-Friendly Rate Control
TSPEC	Traffic Specification
TXOP	Transmission Opportunity
UDP	User Datagram Protocol
VAD	Voice Activity Detection
VBR	Variable Bit Rate
VoIP	Voice over IP
Voice-TFCC	Voice TCP-Friendly Congestion Control
VoWLAN	Voice over Wireless Local Area Network
WAN	Wide Area Network
WLAN	Wireless Local Area Network

Annex A

VoIP Traffic Modeling

Human speech is usually modeled as sequence of alternate talk and silence periods whose durations are exponentially distributed and referred as to ON-OFF model. The presently available codecs with silence suppression mechanism have the ability to improve the speech quality by reproducing speakers back ground using special frames generated during voice inactivity periods. When silence suppression is employed, VoIP traffic generated from one VoIP source can be considered as VBR (Variable Bit Rate) traffic and modeled as an ON-OFF Markov process (Figure 6.1).

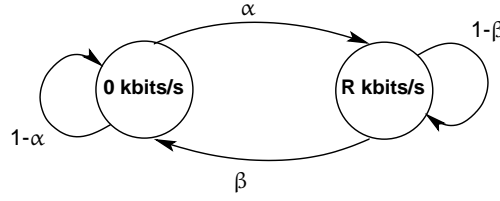


Figure 6.1: ON-OFF modeling of VoIP traffic

Activity and silence periods follow an exponential distributed variable X with average durations of $1/\beta$ and $1/\alpha$, respectively. The density function of an exponential variable X is given by:

$$f(x) = \begin{cases} a \cdot e^{-ax} & \text{for } x > 0, a > 0 \\ 0, & \text{for } x \leq 0 \end{cases} \quad (6.1)$$

where $E[X] = 1/a$ and $\text{var}[X] = 1/a^2$.

The fraction of time that the voice source is in the ON state is $\alpha/(\alpha + \beta)$. ITU-T specification for artificial conversational speech [56] recommend an average talk-spurt of 30.83% and an average silence period of 61.47%. In Brady's model [59], considered mean holding times of the ON and the OFF states are 1 sec and 1.35 sec, respectively. When the source is in the ON state, data is generated at the voice codec rate. Fixed-size packets are generated at a constant interval, the rate is given by (packet size/packet interval). The packet size and the sending rate depend on the voice codec and compression scheme that are used. During the OFF state, no packets are transmitted. For a given codec bit rate R kbits/s, the sending rate of a voice source i that is using silence suppression will be:

$$S_i(t) = \begin{cases} R & \text{when the source is active} \\ 0 & \text{when the source is silent} \end{cases} \quad (6.2)$$

Without silence suppression scheme, packet voice streams can be represented using CBR (Constant Bit Rate) model.

Présentation des Travaux de Thèse en Français

Déploiement à Grande Échelle de la Voix sur IP dans des Environnements Hétérogènes

Dans cette thèse nous étudions le déploiement à grande échelle de la voix sur IP (VoIP) dans des environnements Internet hétérogènes. Dans une première partie de la thèse, nous proposons un nouveau mécanisme de contrôle de congestion pour le trafic de voix sur IP en croissance continue sur l'Internet. Dans une seconde partie, nous développons un modèle analytique pour l'évaluation de la capacité d'un réseau sans-fil 802.11e en terme de communications de voix sur IP. Dans une troisième partie, nous proposons une architecture adaptative permettant d'améliorer la performance de la transmission de la voix sur IP dans des environnement hétérogènes filaires/sans-fil.

1. Introduction

Dans les réseaux téléphoniques à commutation de circuits, le service est garanti durant toute la durée d'une communication mais le nombre de communications simultanées est limité par la capacité du réseau. Dans ces réseaux, l'utilisation efficace des ressources est difficile à atteindre. L'avantage des réseaux à commutation de paquets qui sont à la base de l'Internet est le partage des ressources réseau entre les différents utilisateurs ce qui permet l'utilisation efficace de la bande passante. Cependant, les réseaux IP "best-effort" offrent une remise au mieux des paquets et ne garantissent pas de qualité de service. Le protocole IP (Internet Protocol) [28], au coeur du fonctionnement de l'Internet, assure un service non fiable de délivrance de datagrammes IP. Il n'existe aucune garantie pour que les datagrammes IP arrivent à destination. Certains peuvent être perdus, dupliqués, retardés, altérés ou remis dans le désordre et ni l'émetteur ni le récepteur ne sont informés directement par IP des problèmes rencontrés. IP traite chaque datagramme indépendamment de ceux qui le précèdent et le suivent. Les protocoles RTP (Real-time Transport Protocol) [8] et RTCP (Real-time Transport Control Protocol) [9] permettent respectivement de transporter et de contrôler des flots de données qui ont des propriétés temps-réel (ex. voix sur IP). Le protocole RTCP est basé sur des transmissions périodiques de paquets de contrôle. RTP et RTCP sont des protocoles qui se situent au niveau de l'application et utilisent les protocoles sous-jacents de transport TCP ou

UDP. Mais l'utilisation de RTP/RTCP se fait généralement au-dessus de UDP [29].

Le principe de la voix sur IP est d'appliquer à la voix le même traitement que les autres types de données circulant sur Internet, grâce à la pile de protocoles IP/UDP/RTP. Sont donc transportés des paquets de données constituant de la voix numérisée selon un codec audio donné (ex. G.711, G.723.1, G.729). Le signal numérique obtenu par numérisation de la voix est découpé en paquets qui sont transmis sur un réseau IP vers une application qui se chargera de la transformation inverse (des paquets vers la voix). Les principaux avantages de la technologie du transport de la voix sur IP sont:

- La réduction des coûts des communications téléphoniques
- L'efficacité d'utilisation de la bande passante réseau
- L'intégration des réseaux téléphoniques et des réseaux de données

Au lieu de disposer à la fois d'un réseau informatique et d'un réseau téléphonique commuté (RTC), tout pourrait être fusionné sur un même réseau grâce à la VoIP. Faire basculer différents types de données sur un même réseau permet de simplifier son administration et de réduire les coûts. Les entreprises dépensent énormément en communications téléphoniques, or le prix des communications sur Internet est très réduit en comparaison. En particulier, plus les interlocuteurs sont éloignés, plus la différence de prix est intéressante. De point de vue efficacité, la téléphonie sur IP utilise jusqu'à dix fois moins de bande passante que la téléphonie traditionnelle (transmise à 64 kbps) grâce au partage de la bande passante sur les réseaux IP et aussi grâce aux techniques de codage de la voix sur IP permettant de générer de très bas débits (allant jusqu'à 5.3 kbps pour le codec G.723.1). De plus, la flexibilité de la VoIP permet de faciliter le développement d'applications utilisant la voix et d'autres types de données (ex. vidéo).

Bien que les premières motivations de la transmission de la voix sur IP étaient de réduire les coûts des communications des entreprises et des communications téléphoniques à longue distance (communications internationales), la voix sur IP est actuellement considérée comme la technologie réseau qui va remplacer les réseaux téléphoniques commutés et qui sera la base de la génération future des communications multimédia où un seul réseau de transport des données et de la voix sera utilisé. Depuis quelques années, des outils de voix sur IP sont offerts aux utilisateurs à des coûts de communication de plus en plus réduits (ex. l'outil skype [4]).

Le problème le plus important est la qualité de la transmission de la voix, qui n'est pas encore optimale. La qualité est déterminée principalement par le délai, les variations du délai (gigue), le taux de perte des paquets et le degré de répartition des pertes. En effet, étant données les contraintes d'interactivité, la voix sur IP représente le type le plus sensible du trafic multimédia temps-réel. En particulier, le délai affecte considérablement la qualité de la VoIP. La recommandation G.114 [3] de l'ITU-T (International Telecommunications Union) indique que le délai de bout-en-bout des paquets de voix sur IP ne doit pas dépasser 150 ms (délai dans un seul sens) afin d'obtenir une bonne qualité de la transmission. Par conséquent, les réseaux IP "best-effort" conçus à l'origine pour le transport de flux de données non temps-réel et présentant des délais variables et imprévisibles ne permettent

pas de garantir une bonne qualité de la voix transmise. Deux approches majeures ont été proposées dans la littérature afin de garantir une certaine qualité de service (ex. bande passante, délai, taux de perte) pour les flux multimédia (ex. voix sur IP, vidéo) dans les réseaux IP. La première approche consiste à modifier l'architecture réseau afin de fournir la qualité de service. Principalement, deux modèles d'architecture ont été développés au sein de l'IETF: le modèle de services intégrés IntServ (Integrated Services) [5] et le modèle de services différenciés DiffServ (Differentiated Services) [7]. IntServ est basé sur le protocole RSVP [6] qui permet de réserver, pour chaque flux, les ressources réseau au niveau des routeurs reliant la source et la destination du flux. Cette architecture nécessite la conservation d'informations d'état pour chaque flux au niveau des routeurs et par conséquent la complexité de ce modèle augmente avec l'augmentation du nombre de flux. Le problème de scalabilité de IntServ a poussé l'IETF à considérer l'architecture DiffServ qui offre un modèle plus léger afin d'assurer une qualité de service réseau. Cette architecture est basée sur la classification des flux en différentes classes de service avec des garanties de service qui sont appliquées à l'agrégation des flux Internet d'une même classe. Cependant, la stratégie de DiffServ (différentes files d'attente pour les différentes classes de trafic) présente des problèmes de scalabilité surtout dans le cas d'un grand nombre de flux d'une même classe de trafic. Vu les problèmes de complexité et de scalabilité, les modèles IntServ et DiffServ n'ont pas été déployés à une grande échelle [13]. Dans cette thèse nous ne considérons pas l'approche de qualité de service réseau.

La seconde approche consiste à améliorer la qualité de service sans modifier l'infrastructure réseau. Cette approche se base sur des mécanismes de qualité de service au niveau application. Des implémentations d'applications qui s'adaptent aux variations de l'état de congestion du réseau sont utilisées. Le service fourni par le réseau est mesuré (principalement en terme de délai et de taux de perte des paquets) en utilisant les rapports RTCP [9] par exemple. Ces informations seront utilisées par les applications afin d'adapter convenablement leurs comportements. L'un des principaux mécanismes utilisés pour résoudre les pertes de paquets au niveau application est la FEC (Forward Error Correction) [11][12]. Un exemple simple de FEC consiste à envoyer conjointement avec chaque paquet une copie du paquet précédent de telle sorte que la copie soit utilisée lorsque le paquet original est perdu. L'inconvénient de cette technique est le délai introduit par l'attente de l'information FEC de redondance dans le cas d'une perte de paquet ainsi que l'augmentation d'utilisation de bande passante. Des mécanismes de compensation des variations de délai (gigue) ont été proposés dans la littérature. Dans [10], les auteurs développent un algorithme de tampon (playout buffer) adaptatif qui permet d'atténuer l'effet de la gigue. Cet algorithme présente l'inconvénient de l'ajout d'un certain délai aux paquets. Résoudre les pertes de paquets ainsi que le problème de délai encourus sur les réseaux IP est alors possible. Cependant, l'hétérogénéité de l'infrastructure Internet actuelle rend les solutions de qualité de service du niveau application difficiles à déployer sur l'Internet ubiquitaire. Par exemple, les paquets de voix sur IP peuvent traverser des liens réseau à haut débit (ex. coeur du réseau) et dans ce cas les délais et les taux de pertes engendrés seront très réduits même pendant les piques de trafic, ainsi il y aura pas besoin d'utiliser des mécanismes complexes afin d'améliorer la qualité de la transmission. Cependant, dans le cas des réseaux d'accès, la bande passante est limitée surtout dans le cas des réseaux locaux sans-fil

(WLAN). D'où le besoin de l'utilisation de mécanismes visant à améliorer la performance de la transmission de la voix sur certaines parties du réseau.

Bien que nous ne considérons pas l'approche de qualité de service réseau dans notre thèse, les mécanismes que nous proposons sont compatibles avec cette approche (mécanismes IntServ et DiffServ) et peuvent être utilisés conjointement avec eux afin d'améliorer la performance de la voix sur IP.

2. Motivations et Contributions de la Thèse

Les travaux de recherche s'intéressant à la transmission de la voix sur IP ont débuté en 1977 [1]. Les contributions qui ont suivi se sont focalisées sur la résolution des limitations de performance de la transmission de la voix sur IP liées aux caractéristiques intrinsèques des réseaux IP (délais importants et variables, taux de pertes élevées) conçus à l'origine pour le trafic de données élastiques (non temps réel). Le but était d'assurer une qualité de la voix sur IP équivalente à celle offerte par les réseaux téléphoniques. Les solutions proposées dans la littérature ont étudié le cas d'une seule communication de VoIP et s'appliquent séparément sur différentes communications de voix. Ces solutions se sont intéressées principalement aux mécanismes de compensation de la gigue et de la résolution des pertes de paquets [38]. Cependant, l'aspect du passage à l'échelle de la voix sur IP n'a pas été considéré.

Actuellement, le trafic de la voix sur IP est en pleine expansion vu le déploiement rapide des applications de VoIP. Simultanément, l'hétérogénéité de l'Internet est en augmentation continue, principalement due au succès du déploiement des réseaux locaux sans-fil. L'augmentation considérable du nombre de communications de VoIP ainsi que l'augmentation de l'hétérogénéité de l'Internet ont créé de nouvelles limitations de performance pour le déploiement de la VoIP. La motivation principale de notre travail est de proposer des solutions et des améliorations afin de résoudre ces limitations.

Nous nous focalisons dans cette thèse sur le problème du déploiement à grande échelle de la voix sur IP dans des réseaux IP hétérogènes. Nous étudions le compromis entre l'efficacité et la performance de la transmission d'un grand nombre de communications de VoIP. Notre but est de montrer qu'une bonne qualité de la voix sur IP peut être obtenue par l'utilisation de mécanismes de contrôle adaptatif qui ajustent leurs comportements selon l'état de congestion du réseau afin de:

- Maximiser la performance globale des flux de voix sur IP
- Utiliser les ressources réseau de manière efficace
- Partager équitablement les ressources avec les autres types de flux transmis sur Internet (TCP)

Nous proposons des architectures réseau basées sur des noeuds intermédiaires (passerelles de VoIP) implémentant les mécanismes adaptatifs développés. Les principales contributions de la thèse sont décrites ci-dessous.

2.1. Mécanisme TCP-Amical de Contrôle de Congestion du Trafic de Voix sur IP (Voice-TFCC)

Étant donné le déploiement continu de la voix sur IP, une partie de plus en plus significative du trafic téléphonique mondial est transmise sur Internet. Typiquement, la voix sur IP est transmise en utilisant le service de remise au mieux de l'Internet (best-effort). Le problème c'est que ce trafic "best-effort" n'utilise pas de mécanismes de contrôle de congestion efficace et scalable. Ceci pourra engendrer des effets négatifs sur la performance de la transmission des flux de voix sur IP ainsi que des autres flux partageant les liens réseau avec ces derniers (principalement les flux TCP). Pour ces motivations, nous avons étudié dans [34] les limitations de performance dans le cas d'un grand nombre d'appels téléphoniques à longue distance provenant de sources différentes et transmis à travers un réseau IP "best-effort" entre deux passerelles téléphoniques (Figure 3.3).

En effet, les entêtes protocolaires IP/UDP/RTP représentent une information de taille importante¹ par rapport à l'information de voix compressée et transmise dans les trames audio. Par exemple, si le codec G.729A est utilisé, la bande passante totale requise sera de 24 kbps résultant en un surcoût de 66% (voir Tableau 3.1). Ce qui engendre des effets négatifs pouvant aller de l'utilisation inefficace de la bande passante jusqu'à la congestion du réseau causée par le flux important de paquets de petite taille traversant le réseau IP. En plus, l'absence de mécanismes de contrôle de congestion des flux de VoIP pourra causer un problème d'équité avec le trafic TCP. Deux approches principales ont été proposées dans la littérature afin de résoudre le problème du surcoût protocolaire des paquets de voix sur IP:

- La compression des entêtes IP/UDP/RTP
- Le multiplexage des paquets de voix

Un schéma de compression des entêtes IP/UDP/RTP a été proposé par Casner et Jacobson dans la RFC 2508 [66]. Ce schéma permet de réduire la taille de l'entête de 40 octets à un minimum de 2 octets dans le cas où les champs de somme de contrôle (checksums) de l'entête UDP ne sont pas utilisés. Le mécanisme de compression est basé sur un codage différentiel de l'information contenue dans les entêtes des paquets d'une même session de VoIP. Cependant, ce schéma se base sur les couches inférieures (couche liaison de données) afin de fiabiliser la transmission. Ce qui a limité le déploiement de ce mécanisme.

Concernant la seconde approche, plusieurs schémas de multiplexage de flux RTP ont été proposés [70][75][76][73]. L'hypothèse de base est qu'à un instant donné plusieurs flux RTP sont transmis entre deux passerelles source et destination. Les flux seront alors agrégés pour former un seul flux utilisant une seule entête IP/UDP ou dans une seule entête IP/UDP/RTP. Dans ce dernier cas des mini-entêtes RTP sont utilisées afin de garder les informations contenues dans les entêtes RTP de chaque flux.

Nous avons développé dans cette contribution un nouveau schéma de contrôle de congestion des flux de voix sur IP: Mécanisme TCP-Amical de Contrôle de Congestion du Trafic de Voix sur IP

¹Entête IPv4 de 20 octets, entête UDP de 8 octets et entête RTP de 12 octets

(Voice-TFCC). Voice-TFCC est un mécanisme générique qui permet de:

- Réduire le surcoût des entêtes protocolaires et améliorer l'efficacité de la transmission
- Contrôler le débit généré (en paquets/sec et en octets/sec) des flux de voix sur IP
- Garantir l'équité avec les flux TCP

L'approche développée est d'adapter le protocole de transport ainsi que la manière avec laquelle il interagit avec le réseau afin d'assurer la coexistence des flux de voix et du trafic TCP dans l'Internet. Notre schéma est basé sur un multiplexage dynamique des trames RTP. Il utilise l'équation TCP-friendly (Equation 3.2) pour le contrôle de congestion des flux non adaptatifs. Le mécanisme TCP-amical (TCP-friendly) a été proposé en [78] afin d'avoir un mécanisme de contrôle de congestion propre aux applications multimédia utilisant des protocoles de transport qui ne réagissent pas à l'état de congestion du réseau (protocoles RTP et UDP). Ce mécanisme permet d'estimer le débit moyen d'une connexion TCP comme fraction inversement proportionnelle au délai aller-retour (RTT) moyen et à la racine carrée de la probabilité de perte d'un paquet (square-root-formula). Voice-TFCC utilise le schéma de multiplexage décrit dans [70] où plusieurs flux RTP sont multiplexés en un seul flux UDP. Selon l'information d'estimation de l'état de congestion du réseau², la passerelle source calcule la limite de débit TCP-amical et adapte en conséquent le débit généré en paquets/sec par le multiplexage de paquets (Figures 3.8 et 3.9). Le débit généré en octets/sec est aussi adapté en variant le codage audio utilisé (Figure 3.10). Le mécanisme d'adaptation est effectué selon l'Algorithme 1 décrit dans la Section 3.4.2. Ainsi, le débit généré par les entêtes protocolaires est adapté en réponse à la congestion, tout en respectant les exigences des applications de voix sur IP et éviter une grande variation du débit.

Dans le cas d'une charge importante du réseau impliquant un état de congestion, le mécanisme Voice-TFCC réduit le codage audio utilisé en plus de la réduction du nombre de paquets générés (réalisé par l'augmentation du nombre de paquets RTP multiplexés). Il faut noter que la taille des paquets multiplexés est limitée par la taille maximum MTU (Maximum Transfer Unit). Voice-TFCC peut être appliqué aussi dans le cas d'un seul flux de VoIP, seulement le débit de codage (octets/sec) sera varié dans ce cas (voir le Schéma 3.11).

Dans le cas d'un environnement réseau filaire, le débit en paquets/sec et en octets/sec doivent être contrôlés afin d'améliorer la qualité de transmission. Cependant, dans le cas d'un lien réseau sans-fil (tel que les réseaux IEEE 802.11), il est plus intéressant de réduire le débit en paquets/sec qu'en octets/sec vu le surcoût introduit à la transmission des trames par le protocole d'accès au canal sans-fil.

La performance de notre mécanisme a été évaluée par des résultats analytiques et des expérimentations effectuées sur le réseau PlanetLab [77]. Les résultats obtenus montrent que le schéma Voice-TFCC améliore significativement la qualité de la transmission des flux de VoIP (délai et gigue réduits), permet une utilisation efficace de la bande passante réseau ainsi que l'équité avec le trafic TCP. Le schéma proposé est scalable vu qu'il ne nécessite pas de modifications de l'infrastructure

²Cette information peut être obtenue en utilisant les rapports RTCP

réseau et qu'il utilise un nombre réduit de messages de contrôle (un flux RTCP par plusieurs flux RTP multiplexés). En plus, ce schéma ne nécessite pas de modifications du format standard des paquets RTP. Il est aussi important de noter que Voice-TFCC est inter-opérable avec les mécanismes de qualité de service réseau. Par exemple, le flux de paquets multiplexés pourra être marqué comme étant un flux EF (Expedited Forwarding) pour la différenciation de service et ceci va aider à réduire encore plus les délais au niveau des routeurs Internet.

2.2. Évaluation de la Capacité des Réseaux Sans-Fil IEEE 802.11e en Nombre de Communications de Voix sur IP

Les réseaux locaux sans-fil IEEE 802.11 [85] présentent beaucoup d'avantages par rapport aux réseaux filaires, parmi lesquels la mobilité et la flexibilité, ce qui a facilité leur déploiement dans les réseaux d'entreprise ainsi que dans les "hot spots": centres de conférence, les aéroports, les hôtels, etc. Grâce aux développements importants de la technologie des réseaux locaux sans-fil, ceux-ci présentent actuellement un choix intéressant pour les liens d'accès à l'Internet. Simultanément, la voix sur IP est en pleine expansion grâce à ses avantages d'efficacité d'utilisation de la bande passante, de réduction des coûts de communication et de flexibilité. Nous prévoyons que la convergence de ces deux technologies entraînera une utilisation de plus en plus importante des réseaux sans-fil pour la transmission de communications de voix sur IP. Par conséquent, les réseaux sans-fil auront à supporter un nombre important de communications simultanées de VoIP. Pour ces motivations, nous avons étudié dans cette partie de la thèse (Chapitre 4), la capacité des environnements réseaux locaux sans-fil IEEE 802.11e [106]. Cette capacité est définie en terme du nombre maximum de communications de VoIP simultanées pouvant être supportées par le canal réseau sans-fil. Nous avons étudié, en première étape, l'effet de réduction de la capacité réseau provoquée par les surcoûts des couches protocolaires IP/UDP/RTP/MAC/PHY (Figure 4.5). Ensuite, nous avons développé un modèle analytique pour l'évaluation de la capacité en VoIP sous le mode HCF (Hybrid Coordination Function) du standard 802.11e. Cette capacité est donnée par la somme du nombre de communications donné par l'Equation 4.14 et celui donné par l'Equation 4.16. Ces deux équations modélisent, respectivement, le nombre de communications de VoIP pouvant être transmises durant une période de transmission TXOP (Transmission Opportunity) obtenue après une contention réussie durant le mode d'accès distribué (EDCA-TXOP) et le nombre de communications transmises suite à la réception d'une invitation à transmettre (HCCA-TXOP) pendant le mode d'accès centralisé géré par le point d'accès au canal sans-fil (QAP). Nos résultats de modélisation montrent que l'utilisation des deux modes EDCA et HCCA pour la transmission de la voix améliore considérablement le nombre supporté de communications de VoIP. Utilisant le modèle qu'on a développé, on a étudié l'effet des paramètres de l'application de VoIP (codec audio) ainsi que des paramètres des couches MAC/PHY sur la capacité du réseau (Figure 4.7). On a aussi comparé la capacité des deux modes EDCA et HCCA (Figure 4.10). Les périodes d'accès contrôlé (CAPs) offrent une capacité de VoIP plus importante que celle des périodes d'accès avec contention (CPs). Ceci est dû à l'efficacité réduite du mode d'accès avec contention. Nos résultats montrent que les codecs à bas débit (G.729 et G.723.1) améliorent significativement la capacité du réseau en comparaison avec le codec G.711

pénalisé par son débit élevé (64 kbps).

Nous avons discuté le compromis entre la capacité réseau affectée principalement par les paramètres des couches MAC/PHY et la qualité de la voix sur IP transmise sur le canal sans-fil. Nous avons montré, utilisant le modèle proposé, qu'une qualité optimale de la voix transmise sur un réseau sans-fil (VoWLAN) peut être obtenue à l'aide d'un mécanisme d'adaptation permettant de déterminer le codec audio à utiliser par les applications de VoIP en fonction du nombre de communications de VoIP transmises à un instant donné sur le réseau sans-fil. Les résultats relatifs à cette contribution ont été publiés dans [36].

2.3. Transmission Adaptative de Flux de Voix sur IP dans des Environnements Hétérogènes

L'infrastructure Internet actuelle interconnecte des réseaux hétérogènes. Ces réseaux se basent sur différentes technologies (ex. réseaux câblés/réseaux sans-fil) et présentent différentes caractéristiques (bande passante, délais, taux de pertes). La croissance de l'hétérogénéité de l'Internet est liée principalement au déploiement rapide des réseaux locaux sans-fil. Ce déploiement a créé un intérêt croissant d'utiliser ce type de réseaux pour transmettre des communications de voix sur IP. La transmission de la voix et des données sur un même réseau présente l'avantage de réduction du coût des communications. Cependant, la bande passante du canal sans-fil est limitée et les conditions du réseau sans-fil sont très variables, ce qui cause une détérioration de la qualité de la voix transmise sur ce type de réseaux en particulier dans le cas d'une charge importante du réseau. Plusieurs problèmes de performance doivent être étudiés et résolus afin d'assurer une bonne qualité de la VoIP transmise sur des réseaux hétérogènes.

En effet, dans les environnements réseaux sans-fil, la performance des flux de VoIP peut être dégradée sévèrement à cause de la variabilité, de l'instabilité des conditions réseau et aussi à cause des autres flux de données partageant le lien sans-fil (ex. ftp, Web). L'implémentation de mécanismes de qualité de service était la solution de base proposée pour supporter le transport des flux multimédias, en particulier la voix sur IP, sur les réseaux IEEE 802.11 (standard IEEE 802.11e [106]). Cependant, même avec l'implémentation de mécanismes de qualité de service, les réseaux sans-fil continuent à présenter plus de défis que les réseaux câblés. En plus, des mécanismes d'adaptation sont nécessaires pour atténuer l'effet de l'hétérogénéité réseau sur la transmission de flux de VoIP.

Nous avons étudié l'architecture décrite dans la Figure 5.1. Dans cette architecture le point d'accès est nécessaire afin de transmettre les flux de voix entre un réseau sans-fil local et le réseau Internet. Dans ce cas, le lien d'accès sans-fil présente le goulot d'étranglement pour les flux de voix sur IP. Typiquement, la congestion se produit au niveau du point d'accès surtout dans le cas d'un grand nombre de flux transmis vers le réseau sans-fil. Des pertes de paquets ainsi qu'une augmentation des délais résulteront de cette congestion. En plus, des pertes de paquets dues aux erreurs de transmission sur le canal sans-fil peuvent aussi avoir lieu. Au cas où des applications adaptatives transmettant des données sur le réseau sans-fil ne différencient pas correctement les deux types de pertes de paquets, des problèmes de performance vont se produire: si les pertes dues

aux erreurs de transmission sont considérées comme étant dues à la congestion, les applications vont réduire le débit de transmission et ceci causera une sous-utilisation de la bande passante réseau. D'un autre côté, si les pertes dues à la congestion sont considérées comme étant des pertes du second type, les flux ne vont pas réagir à la congestion et ceci va causer plus de congestion dans le réseau sans-fil. Plusieurs schémas de différenciation entre les deux types de perte ont été proposés dans la littérature [110][111][115]. Ces schémas sont généralement appliqués à des topologies réseau spécifiques et utilisent des statistiques effectuées sur les délais réseau afin de différencier entre les deux types de pertes.

Dans le Chapitre 5 de la thèse, nous avons proposé une architecture réseau, décrite dans la Figure 5.2, qui utilise une passerelle de VoIP localisée au bord d'un réseau d'accès sans-fil et qui est associée à un point d'accès 802.11e (QAP). La passerelle est utilisée pour transcoder les flux de VoIP avant de les transmettre sur le canal sans-fil. Le débit de codage utilisé est déterminé par un mécanisme de contrôle basé sur l'estimation de l'état de congestion du réseau sans-fil. Afin de conserver des niveaux acceptables de qualité de la voix et d'assurer l'équité avec les flux TCP, le mécanisme proposé adapte dynamiquement le codec audio utilisé en utilisant une technique de prévention de la congestion basée sur l'information du délai réseau. Cette technique est inspirée du mécanisme de contrôle de congestion préventif TCP-Vegas [112]. Notre mécanisme permet aussi la différenciation entre les pertes causées par la congestion et celles dues aux erreurs de transmission sur le canal sans-fil (voir Schéma 5.3). L'adaptation appropriée est alors appliquée (réduire le débit de codage/ajouter de l'information de redondance FEC). La performance du mécanisme proposé a été évaluée utilisant le simulateur réseaux NS-2 [37]. Les résultats de simulations montrent que le mécanisme proposé améliore la performance de la transmission de la voix surtout dans le cas de grand nombre de flux de VoIP (Figures 5.4 et 5.5). La capacité du réseau est aussi améliorée grâce à la réduction du débit de codage de la voix en cas de congestion (Figure 5.6). Les simulations montrent aussi que notre mécanisme est plus équitable que le mécanisme TFRC (TCP-Friendly Rate Control) en terme de partage de bande passante entre flux (Figure 5.7). Le mécanisme d'adaptation proposé, ne nécessite pas de modifications du protocole MAC de contrôle d'accès au canal sans-fil 802.11, ce qui facilite son déploiement sur l'infrastructure réseau existante. Cette contribution a été publiée dans [35].

3. Conclusions et Perspectives

Dans cette thèse, nous avons étudié le déploiement à grande échelle de la voix sur IP dans des environnements Internet hétérogènes. Nous nous sommes concentrés sur le cas d'un grand nombre de communications de VoIP transmis simultanément sur les liens du réseau. Nous avons adressé des problèmes en rapport avec trois thèmes de recherche principaux: (i) les mécanismes du contrôle de congestion pour le trafic de VoIP, (ii) l'évaluation de la capacité des réseaux locaux sans-fil IEEE 802.11e en terme de communications de VoIP et (iii) les mécanismes adaptatifs de la transmission de la VoIP dans des environnements Internet hétérogènes avec des liens d'accès sans-fil. Les idées que nous avons proposées dans cette thèse présentent un pas considérable vers un déploiement

efficace, scalable de la voix sur IP qui permet l'amélioration de performance de la transmission des flux de VoIP. Nous résumons dans ce qui suit les contributions de notre travail.

Dans le Chapitre 2, nous avons présenté une vue d'ensemble de la voix sur IP: les mécanismes de base du codage et du transport de la voix sur l'Internet, les caractéristiques des codecs audio standards (G.711, G.726, G.729, G.723.1), les protocoles utilisés pour le transport des flux de voix sur l'Internet (principalement RTP/RTCP), ainsi que les facteurs affectant la qualité de transmission de la VoIP (codecs à bas débits, délai, pertes, gigue). Les questions que nous avons adressées dans ce chapitre sont:

- Le compromis entre l'efficacité d'utilisation de la bande passante et qualité de la transmission de la VoIP principalement affectée par le débit de codage, le délai et le taux de perte des paquets.
- L'effet des caractéristiques du réseau (bande passante, hétérogénéité de l'infrastructure) et de l'état de congestion sur la performance de la transmission de la VoIP.

Dans le Chapitre 3, nous avons étudié le besoin de concevoir un contrôle de congestion pour le trafic de VoIP et nous avons développé un nouveau schéma de contrôle de congestion pour la VoIP appelé Voice-TFCC (Voice TCP-Friendly Congestion Control). Ce schéma combine un multiplexage de flux RTP et un mécanisme de contrôle de congestion TCP-amical afin d'adapter dynamiquement le débit de codage (en nombre et en taille des paquets) des flux de VoIP tout en partageant équitablement la bande passante réseau avec les autres flux transmis sur les mêmes liens (flux TCP). Nous avons évalué la performance de notre schéma à l'aide d'un prototype que nous avons implémenté. Les résultats d'expérimentations effectuées sur le réseau PlanetLab montrent les avantages de l'adaptation du débit de codage. En effet, des améliorations considérables de délai ont été obtenues dans le cas d'une bande passante réduite. De plus, le multiplexage réduit le surcoût des entêtes protocolaires ce qui améliore l'efficacité de la transmission. Appliquer le mécanisme de contrôle de congestion TCP-amical assure aussi l'équité avec les flux TCP. Nous concluons que l'usage de mécanismes de contrôle de congestion pour les flux de VoIP représente une solution prometteuse pour prévenir la dégradation de performance du trafic Internet. Les travaux futurs en rapport avec cette partie de nos contributions peuvent impliquer plusieurs axes de recherche. Ici nous en pointons quelques-uns.

- Étude de l'effet du schéma de multiplexage Voice-TFCC dans le cas de sources VoIP qui utilisent le mécanisme de suppression des silences. Le modèle de source VBR (Variable Bit Rate) devrait être considéré dans ce cas.
- Évaluez plusieurs techniques d'estimation du taux de perte des paquets et leur effet sur le comportement TCP-amical.
- Une extension prometteuse au schéma Voice-TFCC est d'étudier le cas d'une haute charge réseau causé par un grand nombre de flux VoIP qui ne peuvent pas être multiplexés dans un seul flux TCP-amical transmis entre les passerelles source/destination. Dans ce cas, une solution envisageable est de faire passer une partie des flux à travers une troisième passerelle (entre la passerelle source et destination) afin de réduire la charge des liens réseau et d'accomplir

la condition TCP-amical. Pour la sélection de la passerelle, les algorithmes pair-à-pair peuvent être utilisés pour trouver le meilleur chemin Internet entre les passerelles de VoIP en termes de délai et de bande passante. Une autre solution consiste à acheminer les flux de VoIP vers une passerelle qui présente de meilleures conditions réseau pour arriver à la destination. Les protocoles de signalisation tels que SIP (Session Initiation Protocol) pourraient être utilisés pour la communication entre les terminaux et les passerelles de VoIP.

Dans le Chapitre 4, nous nous sommes concentrés sur l'évaluation de la capacité d'un réseau sans-fil IEEE 802.11e en terme de communications de VoIP. Nous avons développé un modèle analytique de la capacité et nous avons présenté des résultats numériques de l'application de ce modèle pour différentes configurations de réseau sans-fil 802.11. Principalement, la capacité des réseaux 802.11e réseaux est réduite à cause de:

- Les entêtes relatives aux protocoles standards (IP/UDP/RTP/MAC/PHY) encapsulant les paquets VoIP de petite taille.
- L'inefficacité inhérente aux protocoles d'accès au canal (MAC) des réseaux sans-fil 802.11 (les intervalles de temps IFS, les mécanismes du backoff, les mécanismes d'acquittement).

Nous avons étudié le cas où les stations sans-fil (QSTAs) peuvent transmettre des trames de voix durant une période de transmission TXOP obtenue après une contention réussie durant le mode d'accès distribué (EDCA-TXOP) ou par la réception d'une invitation à transmettre (HCCA-TXOP) pendant le mode d'accès centralisé géré par le point d'accès au canal sans-fil (QAP). Nos résultats de modélisation montrent que l'utilisation des deux modes EDCA et HCCA pour la transmission de la voix améliore considérablement le nombre de communications de VoIP supportées. Nous avons montré aussi que les périodes d'accès contrôlé (CAPs) offrent une capacité de VoIP plus importante que celle des périodes d'accès avec contention (CPs). Ceci est dû à l'efficacité réduite du mode d'accès avec contention. Nos résultats ont aussi montré qu'en comparaison au codec G.711, les codecs à bas débit (G.729, G.723.1) améliorent considérablement la capacité de la VoIP dans les réseaux 802.11e. Nous avons montré qu'une qualité optimale de la voix transmise sur un réseau sans-fil peut être obtenue à l'aide d'un mécanisme d'adaptation permettant de déterminer le codec audio à utiliser en fonction du nombre de communications de VoIP transmises sur le réseau sans-fil.

Le travail futur concernant cette contribution consistera à proposer un modèle de capacité qui tiendra en compte l'effet d'autres types de trafic (vidéo, ftp) transmis sur le canal 802.11 simultanément avec les flux de VoIP, sur la performance des communications VoIP supportées (délai et perte des paquets). En particulier, l'effet de l'augmentation du temps du backoff causé par les collisions entre les trames 802.11 devrait être étudié.

Une autre extension intéressante à ce travail consiste à concevoir un schéma d'optimisation conjointe de la capacité/qualité dans lequel une passerelle de VoIP interagit avec le point d'accès QAP. Le QAP adapte les paramètres de la couche MAC qui déterminent le nombre de communications de VoIP admissibles (la taille de la supertrame (SF) et l'intervalle du service (SI)), le codec audio qui assurera la qualité optimale est alors déterminé d'après le nombre de communications acceptées.

Dans le Chapitre 5, nous avons étudié la transmission de la VoIP sur des réseaux hétérogènes filaires/sans-fil. En effet, l'augmentation de l'hétérogénéité de l'infrastructure réseau présente un défi principal pour les applications Internet. Les travaux de recherche dans le domaine des réseaux et en particulier la voix sur IP devrait prendre ceci en considération en concevant des protocoles qui s'adaptent dynamiquement et ajustent leur comportement selon l'état de congestion du réseau et aussi par l'utilisation de noeuds intermédiaires permettant d'interconnecter efficacement des réseaux hétérogènes. Nous avons proposé dans cette partie de la thèse, une architecture réseau qui se base sur une passerelle de VoIP localisée au bord d'un réseau d'accès sans-fil et qui est associée à un point d'accès 802.11e (QAP). La passerelle est utilisée pour transcoder les flux de VoIP avant de les transmettre sur le canal sans-fil. Le débit de codage utilisé est déterminé par un mécanisme de contrôle basé sur l'estimation de l'état de congestion du réseau sans-fil. Afin de conserver des niveaux acceptables de qualité de la voix et d'assurer léquité avec les flux TCP, le mécanisme proposé adapte dynamiquement le codec audio utilisé en utilisant une technique de prévention de la congestion basée sur l'information du délai réseau. Notre mécanisme permet aussi la différenciation entre les pertes causées par la congestion et celles dues aux erreurs de transmission sur le canal sans-fil. L'adaptation appropriée est alors appliquée (réduire le débit de codage/ajouter de l'information de redondance FEC). Les résultats de simulations montrent que le mécanisme proposé améliore la performance de la transmission de la voix, surtout dans le cas de grand nombre de flux de VoIP.

Le travail futur lié à cette contribution inclut la réalisation de simulations du mécanisme de prédiction du type de pertes afin d'évaluer sa performance et de proposer d'éventuelles améliorations. Une autre perspective importante consiste à combiner le schéma de contrôle du débit de codage et des pertes avec le modèle de capacité du réseau sans-fil proposé dans le Chapitre 4. Ainsi le débit du flux voix/FEC et les paramètres MAC 802.11e seront choisis de manière optimale afin d'améliorer la qualité des flux de VoIP aussi bien que la capacité du réseau sans-fil. L'usage de l'option RTS/CTS pour la protection d'erreur des trames MAC pourra être considérée aussi. Une décision dynamique de l'utilisation de cette option devrait être prise selon la taille de la trame de voix. Cette taille varie en fonction de l'intervalle du service (déterminé par le QAP) et le débit de codage de la voix (déterminé par l'application de VoIP). Etant donné que les trames de taille réduite sont moins confrontées aux collisions que celles de grande taille, l'option RTS/CTS devrait alors être utilisée uniquement dans le cas de grandes trames afin de ne pas réduire la capacité du réseau sans-fil. Il serait aussi intéressant de définir une interface standard de communication entre couches (couche MAC et couche application) afin de permettre un meilleur contrôle des pertes et de la congestion des flux de voix sur IP dans les environnements réseau sans-fil. Les standards actuels ne proposent pas de telle interface. La prochaine génération de standards réseaux sans-fil 802.11 à l'horizon (en particulier le 802.11n) devrait considérer l'interaction entre les couches protocolaires afin d'améliorer la performance de la VoIP tout en conservant les avantages du modèle d'architecture en couches.

PUBLICATIONS

International Journals

- A. Trad, Q. Ni, and H. Afifi, *Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks*, Published in Lecture Notes in Computer Science Volume 3311 / 2004, Title: Interactive Multimedia and Next Generation Networks.

International Conferences

- A. Trad, F. Munir, and H. Afifi, *Capacity Evaluation of VoIP in IEEE 802.11e Network Environment*, in Proc. of IEEE Consumer Communications and Networking Conference, CCNC'06, Las Vegas, Nevada, USA, January 2006.
- A. Trad, Q. Ni, and H. Afifi, *Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks*, in Proc. of International Workshop on Multimedia Interactive Protocols and Systems MIPS'04, Grenoble, France, November 2004.
- A. Trad and H. Afifi, *TFMC: a TCP-Friendly Multiplexing Control Scheme for VoIP Flow Transmission*, in Proc. of IEEE International Conference on Networking ICN'04, Guadeloupe, March 2004.

National Conferences

- A. Trad and H. Afifi, *Etude des Performances de TCP dans un Environnement Réseau à Très Haut Débit*, Actes du congrès « De Nouvelles Architectures pour les Communications » DNAC'01, Paris, France, November 2001.

RÉSUMÉ

Dans cette thèse, nous nous intéressons au déploiement à grande échelle de la Voix sur IP (VoIP) dans des environnements Internet hétérogènes. Après une description des mécanismes de codage et de transmission de la voix sur l'Internet, nous étudions dans une première partie de la thèse, les limites de performance dans le cas d'une transmission d'un grand nombre de flux de voix sur IP entre deux passerelles téléphoniques. Nous discutons le besoin d'utilisation de mécanismes de contrôle de congestion pour le trafic de voix sur IP qui est en croissance continue sur l'Internet. Nous proposons un nouveau schéma de contrôle de congestion de la voix sur IP. Ce schéma combine le multiplexage de flux RTP et le mécanisme de contrôle TCP-amical (TCP-friendly) afin d'améliorer l'efficacité et la performance de la transmission des flux de voix sur IP et de garantir l'équité avec les autres types de trafic coexistant sur l'Internet. La deuxième partie de la thèse est consacrée à l'étude de la transmission de la voix dans des environnements de réseaux locaux sans-fil IEEE 802.11e. Nous développons un modèle analytique permettant d'évaluer la capacité d'un réseau 802.11e en nombre de communications de voix sur IP en fonction des paramètres du niveau applicatif (codage audio utilisé) ainsi que des paramètres relatifs au canal de transmission sans-fil. Ce modèle peut être utilisé pour ajuster ces paramètres afin d'augmenter la capacité du réseau sans-fil tout en considérant les contraintes strictes des communications interactives de la voix sur IP. Dans la dernière partie de la thèse, nous étudions le cas de la transmission de la voix sur IP dans des environnements Internet hétérogènes constitués en partie par des liens d'accès sans-fil. Nous proposons une architecture basée sur une passerelle de voix sur IP placée au bord du réseau sans-fil. Cette passerelle est utilisée pour adapter les flux de voix aux caractéristiques du réseau sans-fil. Le mécanisme d'adaptation proposé estime dynamiquement l'état de congestion du canal sans-fil et permet la différenciation entre les pertes de paquets causées par la congestion et celles dues aux erreurs de transmission sur le canal sans-fil.

MOTS CLÉS

Voix sur IP, déploiement à grande échelle, codecs audio, contrôle de congestion adaptatif, multiplexage de flux, contrôle TCP-amical, réseaux IEEE 802.11e, capacité réseau, E-Model.

ABSTRACT

In this thesis, we focus on large scale Voice over IP (VoIP) deployment over heterogeneous Internet environments. We first present an overview of VoIP coding and transmission mechanisms. Then, we investigate performance limitations in the case of a large number of VoIP flows transported over wired Internet links between peer VoIP gateways. We address, in the first part of the thesis, the need to design congestion control for the growing class of VoIP traffic and we propose a novel VoIP congestion control scheme. This scheme combines RTP voice flow multiplexing and the TCP-friendly congestion control mechanism in order to improve VoIP transmission efficiency and performance while being fair with coexisting Internet traffic. The second part of this thesis deals with VoIP transmission over IEEE 802.11e wireless LAN environments (WLAN). We develop an analytical model to evaluate the capacity of an 802.11e network in terms of VoIP communications while conditioning on the used audio codec and MAC/PHY parameters. This model can be applied to tune IEEE 802.11e standard parameters in order to increase the WLAN capacity while considering stringent requirements of interactive VoIP communications. In the last part of this thesis we study the case of VoIP transmission over heterogeneous Internet environments where WLAN represents the last-hop Internet access link. We propose an architecture that is based on a VoIP gateway located at the edge of the wired and the wireless network, and used to adapt voice flows according to wireless network characteristics. The proposed adaptation mechanism dynamically estimates wireless channel congestion state and allows the differentiation between congestion loss and loss caused by wireless channel transmission errors.

KEYWORDS

Voice over IP, large scale deployment, audio codecs, adaptive congestion control, flow multiplexing, TCP-friendly rate control, IEEE 802.11e WLAN, network capacity, E-Model.