

Università degli Studi di Padova



XIV Ciclo Dottorato di Ricerca in  
Ingegneria Elettronica e delle Telecomunicazioni  
**Curriculum:** Reti e Sistemi di Comunicazione

# ANALYSIS AND MODELING OF WIRELESS DATA NETWORKS

*Andrea Zanella*

**Coordinatore**

*Prof. Enrico Zanoni*

**Tutore**

*Prof. Gianfranco Pierobon*



Dipartimento di Elettronica e Informatica

AA. 1998–2001



*A mio padre  
e mia madre*



---

# CONTENTS

---

<b>SOMMARIO</b> .....	<b>I</b>
<b>ABSTRACT</b> .....	<b>III</b>
<b>ACRONYMS LIST</b> .....	<b>V</b>
<b>I. INTRODUCTION</b> .....	<b>1</b>
I.1. RADIO SYSTEMS: TOWARDS PERVASIVE CONNECTIVITY .....	1
I.1.A. <i>Fixed wireless Internet access</i> .....	1
I.1.B. <i>A demand for global connectivity</i> .....	2
I.1.C. <i>The last step: Personal Area Networks</i> .....	3
I.2. OPEN ISSUES AND AIMS OF THE THESIS .....	6
I.2.A. <i>Analysis and modeling of Bluetooth radio link</i> .....	7
I.2.B. <i>Scheduling algorithms for piconets and scatternets</i> .....	8
I.2.C. <i>Hybrid UMTS/Bluetooth architecture</i> .....	9
I.2.D. <i>Providing wireless Internet access</i> .....	10
I.2.E. <i>Transport control protocol over wireless link</i> .....	10
I.3. OVERVIEW OF THE THESIS .....	12
<b>II. INDOOR RADIO CHANNELS</b> .....	<b>13</b>
II.1. CHARACTERIZATION OF INDOOR RADIO CHANNEL.....	13
II.1.A. <i>Multi-path channels</i> .....	14
II.1.B. <i>Delay Spread</i> .....	16
II.1.C. <i>Doppler Spectrum</i> .....	18
II.1.D. <i>Threshold Crossing Rate</i> .....	20
II.1.E. <i>Average Fade, Non-Fade period</i> .....	22
II.1.F. <i>Fading process for Bluetooth and DECT radio systems</i> .....	23
II.2. DISCRETE CHANNEL MODELS FOR DIGITAL RADIO LINKS .....	28
II.2.A. <i>A general Markovian model for fading channel</i> .....	29
<b>III. BLUETOOTH RADIO SYSTEM</b> .....	<b>31</b>
III.1. BLUETOOTH: CONNECTING WITHOUT CABLES .....	31
III.2. RADIO SPECTRUM AND MODULATION .....	32
III.3. CHANNEL DEFINITION .....	33
III.4. BLUETOOTH PICONETS .....	34
III.5. PHYSICAL LINK DEFINITION .....	36
III.6. PACKET-BASED COMMUNICATIONS.....	38
III.7. INTER-PICONET COMMUNICATIONS.....	41
<b>IV. BLUETOOTH RADIO LINK: PERFORMANCE ANALYSIS &amp; MODELING</b> .....	<b>43</b>
IV.1. AIMS OF THE STUDY .....	43
IV.2. EXPERIMENTAL ANALYSIS OF BLUETOOTH RADIO LINK .....	44
IV.2.A. <i>Measurement platform and tools</i> .....	44
IV.2.B. <i>Target metrics</i> .....	45
IV.2.C. <i>Statistic confidence &amp; data analysis methodology</i> .....	47
IV.2.D. <i>Experimental results</i> .....	50
IV.3. MODELING BLUETOOTH RADIO LINK.....	52
IV.3.A. <i>Number of states</i> .....	53
IV.3.B. <i>State transition step</i> .....	53
IV.3.C. <i>State transition matrix</i> .....	54
IV.3.D. <i>Error probability vector</i> .....	55

IV.3.E.	Parameters matching .....	56
IV.3.F.	Packet Dropping Probabilities: PDP.....	56
IV.3.G.	Payload Error Probability: $P_{CRC}$ .....	57
IV.3.H.	Validation of the mathematical results.....	59
IV.4.	CONCLUSIONS.....	60
<b>APPENDIX IV-A .....</b>		<b>62</b>
	MARKOV CHAINS MEMORY OVER A REUTILIZATION TIME.....	62
<b>V.</b>	<b>POLLING MECHANISMS FOR BLUETOOTH.....</b>	<b>65</b>
V.1.	STATE OF ART: RELATED POLLING SCHEMES .....	65
V.2.	AN INTEGRATED APPROACH.....	69
V.3.	DUALITY PROPERTY .....	70
V.4.	FAIR BANDWIDTH ALLOCATION .....	71
V.4.A.	Piconet Presence Fraction.....	75
V.4.B.	Scatternet Presence Fraction .....	75
V.4.C.	Fair Share .....	76
V.5.	DESCRIPTION OF ALGORITHM.....	80
V.5.A.	Single piconet with no gateways .....	80
V.5.B.	Scatternet .....	83
V.5.C.	Fairness of the proposed algorithm .....	85
V.6.	EXPERIMENTS AND RESULTS .....	87
V.6.A.	Single gateway in two piconets .....	88
V.6.B.	Piconet with two gateways .....	93
V.7.	CONCLUSIONS.....	94
<b>VI.</b>	<b>A HYBRID ARCHITECTURE FOR INDOOR WIRELESS COMMUNICATIONS .....</b>	<b>95</b>
VI.1.	INTERNETWORKING BLUETOOTH BUBBLES.....	95
VI.2.	CENTRALIZED WIRELESS LOCAL AREA NETWORKS .....	97
VI.3.	SCATTERNET MANAGEMENT .....	99
VI.4.	SIMULATION MODEL & RESULTS.....	102
VI.4.A.	Simulation model.....	102
VI.4.B.	Simulation results.....	102
VI.5.	CONCLUSIONS.....	105
<b>VII.</b>	<b>MEDIUM ACCESS CONTROL IN LMDS SYSTEMS.....</b>	<b>107</b>
VII.1.	INTRODUCTION .....	107
VII.2.	THE C-TDMA PROTOCOL.....	109
VII.2.A.	System structure .....	109
VII.2.B.	Contention policy .....	110
VII.3.	THE MATHEMATICAL MODEL .....	111
VII.3.A.	The offered traffic.....	111
VII.3.B.	Node states .....	112
VII.3.C.	System variables.....	113
VII.4.	EQUILIBRIUM POINT ANALYSIS .....	116
VII.5.	THROUGHPUT AND DELAY.....	122
VII.6.	OPTIMIZATION OF PERMISSION PROBABILITY .....	127
VII.7.	C-TDMA VS. PRMA: PERFORMANCE COMPARISON.....	129
VII.8.	CONCLUSIONS.....	134
<b>APPENDIX VII-A.....</b>		<b>135</b>
	AN OCCUPANCY PROBLEM.....	135
<b>VIII.</b>	<b>TCP OVER WIRELESS LINKS .....</b>	<b>137</b>
VIII.1.	INTRODUCTION: TCP CONGESTION RECOVERY MECHANISMS.....	137
VIII.1.A.	Slow Start and Congestion Avoidance .....	138
VIII.1.B.	Fast Retransmit and Fast Recovery .....	141

VIII.2.	IMPROVING TCP PERFORMANCE OVER WIRELESS LINKS .....	143
VIII.2.A.	<i>End-to-end approach</i> .....	144
VIII.2.B.	<i>Split connection approach</i> .....	144
VIII.2.C.	<i>Link-layer approach</i> .....	145
VIII.3.	TCP WESTWOOD: AN END-TO-END APPROACH .....	146
VIII.3.A.	<i>Overview of TCP Westwood</i> .....	148
VIII.3.B.	<i>TCPW analytic model</i> .....	150
VIII.4.	PERFORMANCE EVALUATION .....	155
VIII.4.A.	<i>Ideal error-free connection</i> .....	157
VIII.5.	A SIMPLIFIED ANALYTIC MODEL .....	159
VIII.6.	CONCLUSION .....	161
<b>IX.</b>	<b>CONCLUSIONS</b> .....	<b>163</b>
IX.1.	LIST OF PUBLISHED PAPERS .....	167
<b>X.</b>	<b>REFERENCES</b> .....	<b>169</b>



---

# SOMMARIO

---

L'ultimo periodo è stato caratterizzato da una incredibile crescita della rete Internet, in termini di utenza e servizi forniti, e dal proliferare di dispositivi elettronici personali sempre più economici. Queste tendenze stanno generando una crescente domanda per una rete globale e pervasiva, in grado di garantire intercomunicabilità tra i dispositivi elettronici personali come pure accesso mobile e ubiquo a Internet. In tale contesto, le tecnologie radio assumono un ruolo di primaria importanza, poiché consentono di realizzare la comunicazione dati, a diversi livelli, in modo economico. Di conseguenza, esiste un crescente interesse nella comprensione delle reali potenzialità e dei limiti di tali tecnologie, al fine di identificare i possibili problemi e sviluppare soluzioni efficienti.

La tesi tratta delle problematiche concernenti l'analisi e la definizione di modelli di reti radio personali e locali. Il testo considera aspetti legati alla gestione di reti non cablate a differenti livelli, dalle reti ad estensione personale (PAN) fino ai sistemi cellulari locali e globali.

L'analisi si concentra inizialmente sulla tecnologia Bluetooth, un'interfaccia radio emergente che si pensa assumerà un'importanza primaria nel campo delle reti personali. Al fine di valutare le prestazioni ottenute dai diversi formati di pacchetto dati previsti dallo standard, si sono eseguite una serie di misurazioni sul campo, raccogliendo le statistiche di trasmissione attraverso un collegamento radio Bluetooth. Dai dati raccolti, quindi, si ricava un semplice modello matematico per il collegamento radio che cattura gli effetti aggregati di rumore, affievolimento del segnale e interferenza.

Successivamente, si considera il problema della gestione del traffico (*scheduling*) nelle *piconet* e nelle *scatternet*, le due strutture di rete basilari definite da Bluetooth. In questo contesto, si definisce un principio di dualità tra i ruoli svolti dalle unità di controllo (*master*) e le unità di collegamento tra piconet (*gateway*). Sulla base di tale principio è quindi proposto un algoritmo di scheduling che mira a realizzare una distribuzione equa delle risorse tra tutte le unità connesse in una scatternet. Simulazioni

al calcolatore provano l'effettiva capacità dell'algoritmo di allocare in modo equo la banda di trasmissione tra tutte le unità attive in una scatternet e di adattarsi alle variazioni del traffico offerto dai vari nodi, in modo dinamico.

Le potenzialità delle piconet e delle scatternet Bluetooth possono essere ulteriormente estese utilizzando altre tecnologie radio, con maggiore raggio di copertura, per interconnettere scatternet isolate e separate. In questa tesi si propone di integrare Bluetooth con UMTS in una architettura ibrida in grado di realizzare la comunicazione di dati senza cavi in ambienti interni. Lo studio si focalizza su una topologia a stella, in cui la comunicazione dei dati avviene solamente tra una unità centrale e alcuni nodi periferici. Si considera quindi il problema della distribuzione equa delle risorse tra le unità periferiche e si analizzano le prestazioni del sistema attraverso simulazioni al calcolatore.

Successivamente, l'attenzione si concentra sui sistemi per la distribuzione locale di servizi digitali tramite tecnologie radio multiutente. In questo contesto, si analizza il protocollo Contention–Time Division Multiple Access (C–TDMA), proposto per gestire l'accesso degli utenti del sistema cellulare al canale uplink, ovvero verso la stazione base. Il C–TDMA è descritto attraverso la teoria delle catene Markov, mentre le sue prestazioni, in termini di ritardo e traffico medi raggiunti, vengono valutate con il metodo dell'analisi al punto di equilibrio e verificate tramite simulazioni al calcolatore. I risultati indicano che, nei tipici scenari considerati, il protocollo C–TDMA realizza alti valori di traffico con un ritardo limitato. Inoltre, il confronto con il noto protocollo per l'accesso multiplo a prenotazione di pacchetto (PRMA) mostra che i due protocolli raggiungono prestazioni del tutto comparabili.

Infine, si considerano problematiche relative alla trasmissione dei protocolli TCP/IP su reti radio. In particolare si presenta e analizza il protocollo TCP Westwood (TCPW), una nuova versione del protocollo di trasporto TCP che mira a ridurre l'impatto della perdita sporadica di un segmento accelerando la procedura di recupero. In tale contesto si sviluppa un modello analitico di TCPW che viene quindi convalidato da simulazioni al calcolatore. Il modello è poi utilizzato per comparare TCPW con il classico TCP Reno per diversi valori di capacità minima lungo la connessione, tempo di andata e ritorno, capacità del buffer e probabilità di errore sul pacchetto.

---

# ABSTRACT

---

The recent period has been characterized by the dramatic growth of the Internet, in terms of users and provided services, and the proliferation of economic personal electronic devices. These trends are flowing together into a rising demand for a global and pervasive networking, which would be able to provide interoperability among personal electronic devices as well as ubiquitous and mobile Internet access. In this context, radio technologies are assuming a primary role by virtue of their capability of providing cost-effective seamless data communication at different levels. There is, therefore, a growing interest in understanding the actual potentialities and limits of such technologies, in order to identify the potential problems and develop efficient solutions.

The thesis deals with the issues concerning the analysis and modeling of such wireless data networks. The dissertation covers aspects related to wireless communications at different levels, from personal area networks up to local and global cellular systems.

We start our analysis considering the Bluetooth technology, an emerging radio interface that is expected to play a major role in the field of Personal Area Networks. We perform a series of measurements in order to investigate the potential performance tradeoff among the different data packet formats supplied by Bluetooth. The collected data are, then, elaborated to derive a simple mathematical model of the Bluetooth radio link that captures the aggregate effects of fading, interference and noise.

Successively, we address the issue of scheduling in piconets and scatternets, the basic structures which Bluetooth builds its networking capability on. We define a duality principle between the roles played by masters and gateways. On the basis of such duality, we define a scheduling algorithm that aims to achieve max-min fair capacity distribution among all the units in a scatternet. The algorithm, tested through computer simulations, proves to be traffic-adaptive and achieve fair allocation of bandwidth to units.

The potentiality of Bluetooth piconets and scatternets may be further extended by using other radio technologies with higher coverage range to interconnect scatternet

islands. In this thesis we investigate a hybrid architecture of Bluetooth and UMTS that aims to provide indoor wireless/mobile communications. We focus on a centralized topology, in which data communication occurs between a central node and many peripheral nodes only. Therefore, we address the issue of fair capacity allocation and investigate the performance achieved by such a topology through computer simulations.

Afterwards, our analysis focuses on wireless point-to-multipoint technologies for providing wideband digital services and Internet access. In this context we analyze the Contention-Time Division Multiple Access (C-TDMA) protocol, an access protocol for the up-link channel of cellular multi-user systems. C-TDMA is described by using classical Markovian approach, while performance evaluation of C-TDMA has been made in terms of throughput and delay by using the equilibrium point analysis (EPA) and computer simulations. The results indicate that C-TDMA yields high throughput values with a limited delay in typical cellular scenarios. The protocol is further compared with the Packet Reservation Multiple Access (PRMA) protocol, showing very similar performance.

Finally, we address the issue concerning the transmission of TCP/IP protocols over wireless networks. In the thesis, we present and analyze TCP Westwood (TCPW), a new version of the TCP protocol that aims to reduce the impact of sporadic segment loss by accelerating the successive recovery phase. An analytical model for TCPW connection is proposed and validated through computer simulations. The model is further used to compare TCPW and the classic TCP Reno over a single-bottleneck connection with different values of bottleneck capacity, round trip time, buffer size, packet error probability.

---

## ACRONYMS LIST

---

AC	Access Code field
ACK	ACKnowledgement
ACL	Asynchronous Connection–Less link
AMA	Active Member Address
AODV	Ad–hoc On–demand Distance Vector
ARQ	Automatic Repeat request
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BS	Base Station
BSC	Binary Symmetric Channel
BT	Bluetooth
BWE	BandWidth Estimate
CAC	Channel Access Code
cdf	Cumulative Distribution Function
CDMA	Code Division Multiple Access
CN	Central Node
CPE	Customer Premise Equipment
CRC	Cyclic Redundancy Check
C–TDMA	Contention Time Division Multiple Access
CWLAN	Centralized Wireless Local Area Network
cwnd	congestion window
DAC	Device Access Code

DACK	Duplicate Acknowledgment
D-AMPS	Digital-Advanced Mobile Phone Service
DARPA	Defense Advanced Research Projects Agency
DC	Direct current
DECT	Digital Enhanced Cordless Telecommunications
DH1	High-rate Data packet, 1-slot long
DH3	High-rate Data packet, 3-slot long
DH5	High-rate Data packet, 5-slot long
DM1	Medium-rate Data packet, 1-slot long
DM3	Medium-rate Data packet, 3-slot long
DM5	Medium-rate Data packet, 5-slot long
ELN	Explicit Loss Notification
EPM	Exhaustive Pseudo-cyclic Master
ERR	Exhaustive Round Robin
ETSI	European Telecommunications Standards Institute
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FH	Frequency Hopping
FHSS	Frequency Hopping Spread Spectrum modulation
FM	Frequency Modulation
FSMC	Finite-State Markov Chain
FTP	File Transfer Protocol
FWA	Fixed Wireless Access
GFSK	Gaussian-shaped Frequency Shift Keying

---

GMSK	Gaussian-shaped Minimum-frequency Shift Keying
GSM	Global System for Mobile communications
HEAD	Header field
HEC	Header Error Check
IAC	Inquiry Access Code
IEEE	Institute of Electrical and Electronic Engineers
IMT-2000	International Mobile Telecommunication union
IP	Internet Protocol
IPU	Inter-Piconet Unit
ISI	Inter-Symbol Interference
ISM	Industrial Scientific Medical
IT	Information Technology
I-TCP	Indirect-Transmission Control Protocol
L2CAP	Logical Link Control and Adaptation Protocols
LAN	Local Area Network
LAP	Lower Address Part
LMDS	Local Multipoint Distribution Services
LOS	Line of Sight
LRR	Limited Round Robin
LWRR	Limited and Weighted Round Robin
MAC	Medium Access Control
MC	Markov Chain
NACK	Negative ACKnowledgement
PAN	Personal Area Network
PAYL	Payload field

PDA	Personal Digital Assistant
pdf	Probability Density Function
PDU	Packet Data Unit
PLL	Phase–Locked Loop
PMA	Parked Member Address
PN	Pseudorandom Noise; Peripheral Node
PP	Packet Pair
PRMA	Packet Reservation Multiple Access
PRR	Pure Round Robin
psd	Power Spectrum Density
RF	Radio Frequency
rms	root mean square
RTO	Retransmission Timeout
RTT	Round Trip Time
rwnd	receiver–side window
SACK	Selective Acknowledgment
SAR	Segmentation And Reassembly
SCO	Synchronous Connection Oriented link
SIG	Special Interest Group
SMSS	Sender Maximum Segment Size
SNR	Signal–to–Noise Ratio
ssthresh	slow start threshold
TCP	Transmission Control Protocol
TCPR	Transmission Control Protocol, Reno version
TCPW	Transmission Control Protocol, Westwood version

TCR	Threshold Crossing Rate
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
UCLA	University of California, Los Angeles
UMTS	Universal Mobile Telecommunications System
VLSI	Very Large Scale Integration
W-CDMA	Wideband-Code Division Multiple Access
WLAN	Wireless Local Area Network
WSSUS	Wide-Sense Stationary Uncorrelated Scattering



---

# I. INTRODUCTION

---

*This chapter introduces the topics considered in the thesis and motivates the study carried out. The state of art in the field of wireless data networks is briefly presented, together with the most significant trends that have characterized its recent evolution. Furthermore, we discuss the open issues concerning various aspects of wireless networking and we introduce the original contributions given by the thesis. Finally, we describe the structure of the thesis and list the contents of each chapter.*

## **I.1. Radio systems: towards pervasive connectivity**

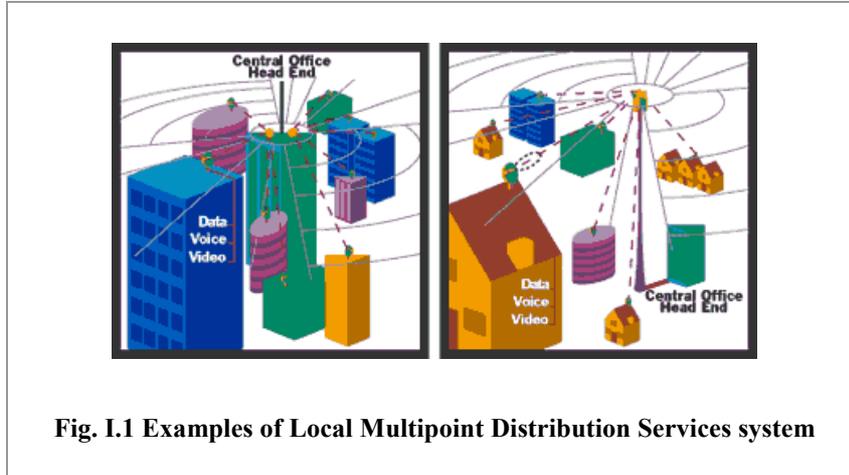
The last decades have been witness of a dramatic evolution in the communications' world that is involving not only the communication means, but even the way in which the *concept* of communication is intended. As a matter of fact, the “possibility” of communicating through the traditional telephone and data networks is progressively turning to a users “demand” for a global and ubiquitous connectivity service that involves not only the limited sphere of telephony, but the entire world of data-communications.

In this context, radio technologies assume a primary role, by virtue of their capability of providing wireless and mobile connectivity at personal level, as well as on a much wider scale. Therefore, researchers are frantically working to develop the radio technologies, standards and protocols that will satisfy the increasing demand for global connectivity. The result is the proliferation of many different radio systems, in part complementary and in part competitive one another, that aim to satisfy the users requirement at every level. In the following, we introduce some of these technologies together with the reasons that have promoted their development.

### **I.1.A. Fixed wireless Internet access**

One of the most significant trends in the panorama of modern communications has been the impressive growth of the Internet. In the space of few years, the Internet has dramatically enriched the number and typologies of services it provides, breaking

through the borders of business and entertainment to become an essential tool in daily life. As the importance of the Internet grows, the demand for pervasive Internet access becomes more urgent.



**Fig. I.1 Examples of Local Multipoint Distribution Services system**

Even though cable (coaxial and optical fiber) remains the most common solution for providing Internet access, cellular radio systems offer a viable complementary solution in some particular scenarios, by virtue of fast deployment, minimum infrastructure impact within cities and cost effectiveness in rural or sparse populated areas. This consideration has promoted the development of fixed wireless access (FWA) solutions and, in particular, of local Multipoint Distribution Services (LMDS) systems. Such systems may provide wide band digital services and Internet access to many static subscribers, distributed on an cell area of few kilometers (Fig. I.1). Therefore, LMDS systems may give a significant contribution to the goal of providing pervasive Internet access.

### **I.1.B. A demand for global connectivity**

The evolution of the Internet has been going together with the introduction and wide diffusion of cellular and mobile communication systems, like Global System for Mobile communications (GSM), Digital-Advanced Mobile Phone Service (D-AMPS), and IS-95 [56][83][82]. Such cellular systems have given users the possibility of communicating from everywhere and at anytime, without the bother of plugging into a wall. This possibility, originally intended for voice services, is rapidly changing into a request for wireless and global data-communication services. In particular, users want

to exchange data, connect to the company network, or access the Internet without the need for cables and wherever they are, at work, at home or on the move.

Such requirements have led, on the one hand, to the diffusion of Wireless Local Area Network (WLAN) technologies ([1][42][47]) designed to offer high-speed wireless connectivity in indoor environments and, on the other hand, to the development of third generation mobile systems, mainly intended to provide globally available low-speed data services.

One of the most promising technologies, in this context, is the universal mobile telecommunication system (UMTS) [26], which is being standardized within the international mobile telecommunication union (IMT-2000) framework. UMTS will build on and extend the current second generation cellular technology by providing enhanced capacity, support of data traffic and a greater variety of services. There are mainly two air interface technologies for UMTS: frequency division duplex (FDD) and time division duplex (TDD) [30]. UMTS-FDD relies on wideband-code division multiple access (W-CDMA) technology and will be deployed in outdoor macro-cellular or micro-cellular communication environments. UMTS-TDD adopts a combination of code division multiple access (CDMA) and time division multiple access (TDMA) technologies, and will be deployed in indoor pico-cellular communication environments. It allows asymmetric radio resource allocation between uplink and downlink and higher bit rate services than FDD and is expected to be used in indoor environments and hot spots.

### **I.1.C. The last step: Personal Area Networks**

The request for a pervasive and global connectivity is further fuelled by a new demand for interoperability among personal electronic devices [69]. Progress in microelectronics and very large scale integration (VLSI) technology has fostered the development and commercialization of many electronic devices, like computers, laptops, cellular phones, palmtops, personal digital assistants (PDAs), digital cameras, printers, and so on. These products are, in general, complete and self-sufficient. Notwithstanding, their interconnection may further enhance their capabilities, enabling

the sharing of data and information, like addresses, files, notes, as well as physical equipments, like keyboards, displays, memory.

A network formed by the interconnection of devices a user usually carries on his person, like cellular phone, headset, PDA, and laptop, is often referred to as personal area network (PAN) [69]. Typically a PAN is centered on the individual himself and moves with him. However, the PAN can expand and contract dynamically, depending on the user's needs, and can include devices that are not carried along with the user, like office printers, sensors located in a room, or wall repeaters for accessing the Internet or company Local Area Network (LAN). Moreover, a PAN may occasionally connect devices of other users' PANs, in order to enable the sharing of any kind of information.

Since a PAN is expected to move along with the user, there is the need of using technologies that are "legally allowed" in the most part of the world. For this reason, the radio interfaces that have been proposed for PANs typically operate in the industrial scientific medical (ISM) frequency bands, which are worldwide available and free by government license. In general, these radio systems are capable of providing connectivity over short distances (possibly up to 100 m) at up to few megabits per second in link speed.

Besides the basic cables replacing functionality, these technologies may enable a large number of other, fascinating applications. It may be possible, for instance, checking and sending e-mails from the car or on the train, connecting the laptop to the Internet through a GSM (or possibly an UMTS) cellular phone, which may be even held into a briefcase or a pocket. Analogously, a laptop may access the Internet from public areas, like airport lounges, libraries, cafeterias, as long as they are equipped with access points supporting the same PAN radio interface. Moreover, the laptop may recognize the proximity of the corresponding fixed computer at home or in the office, and may automatically setup a connection and synchronize the contents of some folders. In the same way, a palmtop and cellular phone may establish a connection in order to refresh contacts list and agenda. Furthermore, laptops supporting PAN functionalities may setup ad-hoc networks to enable the direct exchange of information (documents, presentations, images, messages, business cards, contacts info) among people attending a meeting, without requiring a temporary internet (IP) address to the hosting network.

The application scenarios that may be open by PAN technologies include even some futuristic solutions, like the so-called “smart house”. This scenario envisions a fully networked house in which doors, lights, heaters, household appliances, and virtually any other electronic device in the house are equipped with a common radio interface and can interact dynamically with user. For instance, when the user approaches the house, a sentinel sensor may recognize his/her presence (maybe checking an identification code burned into an electronic key ring) and order the front door to open automatically, while lights switch on, heater sets the room temperature to the programmed value and stereo tunes to the preferred station.

This rapid overview of the applications that may be enabled by PAN technologies gives a hint of the huge potentiality of such systems and motivates the great interest they are gathering on industrial and scientific community.

### ***Bluetooth: a promising radio interface for PANs***

A technology that is expected to play a major role in the field of personal area networks is the new radio interface called Bluetooth [35][36][37][31][70]. Bluetooth provides a reliable data transmission in unlicensed 2.4GHz band, by using a fast frequency hopping technique and a simple retransmission scheme at baseband level. The protocol supports both synchronous voice connections and asynchronous, symmetric and asymmetric, data connections. Networking in Bluetooth is based on an elementary entity called *piconet*. A piconet is established when two or more units connect each other, sharing a common frequency-hopping channel. In a piconet, one unit acts as *master* while the others become *slaves*. The master unit, among other tasks, performs channel access control in the piconet, on the basis of a centralized polling scheme.

A Bluetooth piconet can be viewed as a basic personal area network. The limited number of the devices connected, their spatial proximity, and the centralized polling scheme adopted to control medium access are consistent with the characteristics of such kind of networks. Furthermore, several piconets can be linked together in ad-hoc structure, named *scatternet*, to allow communication in flexible configurations. Once all users are organized in piconets and piconets are linked in a unique scatternet,

communication can take place seamlessly and interaction among devices can be enabled. Bluetooth, thus, may be used also to bridge devices in one's own personal network to the external world, achieving the desired global connectivity.

Bluetooth was initially proposed by Ericsson as a short-range cable replacement for linking electronic products, like cellular phone, headset, mouse, and computer. However, the prospect of integrating Bluetooth radio interface in any electronic device is turning the technology in one of the most promising solutions for wireless PAN. As a matter of fact, the IEEE 802.15 Personal Area Network Working Group, formed in the 1999, has made Bluetooth the foundation for a range of consumer network products, most of them portable. Bluetooth is now promoted by a consortium of leading companies in the sector of information technology (IT). The consortium, named Special Interest Group (SIG) [81], was formed in February 1998, by Ericsson, Nokia, IBM, Toshiba and Intel. This group was further extended in December 1999 with 3Com, Lucent, Microsoft, and Motorola. In addition to these nine promoter companies, more than two thousand companies have joined as adopters of the Bluetooth technology.

## **I.2. Open issues and aims of the thesis**

This brief overview of recent evolutions in telecommunications' world highlights a general trend towards the creation of a global and ubiquitous networking. The attainment of such a goal requires the organization of several different wireless technologies in hybrid architectures, which would be able to provide pervasive and distributed Internet access, as well as interoperability among personal electronic devices. In this context, wireless PAN technologies, and Bluetooth in particular, assume a great importance since they represent the conjunction point between final user and digital community.

In the thesis we address some important issues concerning wireless networking. The study ranges from the modeling of physical radio channels to the analysis of transmission control protocols over lossy links. In particular, we deal with the performance analysis, modeling and management of personal area networks based on the Bluetooth standard. Then, we investigate a possible hybrid architecture of Bluetooth and UMTS-TDD that takes advantage of the complementary features of the two

technologies to provide indoor wireless/mobile communications. Successively, we investigate a medium access protocol for uplink radio channel in local multipoint distributed services systems. Finally, we consider the issues related to the transmission of internet protocols, in particular of Transmission Control Protocol (TCP), over wireless links. In the following we introduce the topics considered in the thesis along with the reasons that have motivated the study and the original contributions which have been rendered.

### **I.2.A. Analysis and modeling of Bluetooth radio link**

A primary interest, approaching a new technology, is on performance aspects. In the specific case of Bluetooth, the standard provides up to six packet formats for asynchronous data traffic that differ for time duration, data capacity and error protection. In order to gain a deep insight into the system behavior is, then, necessary to analyze the impact of the different packet formats on the overall system performance.

Technical literature counts some papers that partially address this topic, dealing with the problem of segmentation and reassembly (SAR) of high-layer data units into smaller baseband radio packets in an efficient way. In such studies, however, performance is usually evaluated referring to the nominal value of packet capacity and, therefore, ignoring the effects due to physical radio channel.

The approach adopted in the thesis, instead, aims to determine the actual efficiency achieved by various packet formats in real operating conditions. For this purpose, we perform a large series of experiments, in typical real-world environments, and collect the statistics of the radio transmission. Measurement trials are, then, elaborated in order to extract performance indexes, like packet error probabilities and average throughput, that characterize different packet formats.

Another interesting topic, concerning the performance analysis of radio systems, is the definition of simple mathematical models that can be used to simulate the radio channel behavior. Classical radio channel models are based on simple Markov chains (MC), with a limited number of states. Usually there is a trade-off between simplicity of the model (in terms of implementation and execution time) and accuracy of the results (in terms of matching between simulated and real channel behavior). The greater is the

number of states considered, the more complex is the model but the better is the correspondence between simulations and real behavior.

In the thesis we address the issue of defining a simple mathematical model for Bluetooth radio link. The model stems from the analysis of data collected during the measurement campaign and, thus, aims to capture the aggregate of real-world effects, like noise, interference and fading.

### **I.2.B. Scheduling algorithms for piconets and scatternets**

Besides radio channel conditions, another factor that drastically influences the performance perceived by higher layer applications is the polling policy adopted at medium access control (MAC) layer. Even though polling schemes have been widely investigated in literature, the peculiarities of Bluetooth make classical approaches rather inefficient. A main constraint in piconet scheduling arises from the lacking in information on the queue length at each slave that is available at the master. Slaves can communicate only when directly addressed by a master transmission. Therefore, if the traffic between master and slave is not symmetric, the master may be forced to send empty packets to solicit slave transmissions or, vice versa, the slave may reply with empty packets to acknowledge packets reception. Since the applications proposed for such systems typically produce asymmetric and bursty traffic, a non-optimal scheduling policy may result in considerable performance losses.

The situation is even more critic in scatternets, in which traffic is forwarded from a piconet to another by some inter-piconet units (IPUs) that act as gateways. The gateway role can be played, in particular, by a slave unit that is connected to two or more master units and switches from one to another on a time division base. There is, then, the need for synchronizing master and gateway in a piconet, in order to avoid bandwidth wastage that may occur when a master polls a gateway that is listening to a different piconet.

Most of the published works on Bluetooth scheduling algorithms has been focused on the single piconet. On the other hand, works dealing with scatternets are usually concerned with the actual formation of Bluetooth piconets and scatternets, ignoring scheduling issues.

In this thesis, we propose a global approach to the scheduling problem. This approach stems from the observation that the roles played by master in a piconet and gateway in a scatternet are tied by a duality relationship. Thus, we first extend the concept of max–min fairness to a scatternet structure. Then, we define a totally distributed scatternet scheduling algorithm that provides an integrated solution for both intra– and inter–piconet scheduling. The algorithm aims to achieve fair distribution of both piconet capacity among slaves and gateway capacity among masters it belongs to, without requiring any information exchange among different piconets.

### **I.2.C. Hybrid UMTS/Bluetooth architecture**

Scatternets allow interconnecting several piconets in wider networks, enhancing in such a way their networking capability. Unfortunately, practical considerations limit the maximum extension a scatternet can assume. Indeed, with the increasing of the scatternet size also the complexity of creating, maintaining and managing the scatternet increases, while performance, in terms of average connection latency and throughput, decreases. However, using other radio technologies with higher coverage range to connect scatternets islands may further extend the network.

In this thesis we consider a solution based on the unlicensed UMTS–TDD system that has been specifically proposed to provide voice and data connection in indoor environments. In particular, we focus on a centralized topology, in which data communication occurs between a central node and many peripheral nodes only. In such architecture, Bluetooth provides radio access interface to the network, while UMTS connects isolated Bluetooth islands to the central node. The architecture proposed may result suitable in many scenarios, in which the low cost objective for the provided service is more important than performance aspects. Besides defining the centralized wireless local area network architecture and describing a possible realization based on Bluetooth and UMTS–TDD technologies, we address the issue of fair capacity allocation and provide some simulation results for the topology considered.

### **I.2.D. Providing wireless Internet access**

Wireless point-to-multipoint technologies offer a method of providing high-capacity local access that may result more advantageous than classic wireline solution. As a matter of fact, Local Multipoint Distributed Services (LMDS) wireless systems are characterized by low entry and deployment costs, great speed and easiness of deployment with minimal disruption to the community and the environment, and cost-effective network maintenance and management. Thus, LMDS can be used by competitive service providers to deliver services, like digital voice, data, Internet, and video stream, directly to end users. The radio part of LMDS consists primarily of base station and customer premise equipments (CPEs). The customer-premise configurations may vary widely from vendor to vendor. In particular, the CPE may attach to the network using time-division multiple access (TDMA), frequency-division multiple access (FDMA), or code-division multiple access (CDMA) methodologies.

In the thesis we focus on the access protocol proposed by the European consortium called CRABS (Cellular Radio Access Broadband Services) that aims to provide digital interactive services via microwave cellular radio [23]. The protocol, named Contention-TDMA [65][67][66], is essentially based on a TDMA scheme with reservation, which it adds a contention phase to. The contention mechanism is used by customers to reserve a transmission-slot on the uplink channel. Once the reservation is obtained, transmission can take place with no risk of collisions.

The goal of the thesis is to give a mathematic description of the C-TDMA protocol and investigate the system stability and performance under various operating conditions. Furthermore, to investigate potential advantages of C-TDMA with respect to other reservation-based protocols, a performance comparison is made with the Packet Reservation Multiple Access (PRMA) [59][58], a well-known access protocol which has shown optimal performance in radio cellular systems.

### **I.2.E. Transport control protocol over wireless link**

Even though wireless systems appear suitable for supporting mobile and pervasive Internet access, transmission of Internet protocols over radio links may be very inefficient. It is well known, indeed, that TCP is tuned to work well in wired networks,

where packet losses and unusual delay are primary due to congestion. Unfortunately, communication over wireless links may suffer from sporadic high bit error rates that produce packet losses not related to congestion. These events may trigger congestion reaction mechanisms on TCP sender [74], resulting in an unnecessary reduction in end-to-end throughput and, hence, sub-optimal performance [18].

A method proposed to alleviate these problems consists in hiding the unreliability of the wireless link from TCP sender by using local retransmissions and forward error correction (FEC) schemes [22][20]. In this way, the lossy link appears as a reliable link with a reduced effective bandwidth. An example of such a solution is given by the Bluetooth system. Bluetooth MAC layer, indeed, provides a reliable data transfer service to upper layers by means of an automatic retransmission query (ARQ) protocol that automatically retransmits radio packets until they are correctly received and acknowledged.

Unfortunately, the TCP sender may not be fully shielded from wireless losses. Indeed, the link layer retransmissions could generate sporadic long delay on segment delivery, causing the TCP sender timer to expire. These spurious timeouts trigger unnecessary segment retransmission and start the congestion control mechanisms, leading to a waste of available capacity in the wireless link and a significant degradation of end-to-end throughput. Another problem derives by the low radio link capacity that could produce a buffer overflow at the interface between fixed and radio parts, resulting in a loss of performance. Therefore, link layer solutions can solve the problem of spurious retransmissions and connected performance losses only partially.

Another possible approach to the problem is at end-to-end layer. Such kind of solutions consists in modifying the TCP protocol itself in order to consider wireless links issues. TCP Westwood (TCPW), a new TCP version recently proposed by the University of California, Los Angeles (UCLA), pursues the end-to-end approach. The basic idea of this protocol is to use the feedback information provided to sender by the flow of acknowledgments, in order to estimate the actual bandwidth available along the TCP connection. This estimation is then used to speed up the recovery phase after a packet loss event.

In this thesis we aim to get a deep insight into the bandwidth estimation mechanism adopted by TCPW, and to investigate the performance of the TCPW protocol to varying of the systems parameters. To reach such goals, we define an analytic model that captures the most interesting peculiarities of the protocol and reproduces the simulation results with enough accuracy. Beside the detailed model, we propose a simplified model that, against a much easier implementation, yields very good results. Models are then utilized to perform a comparison with the TCP Reno version, in order to determine the actual advantages yielded by the bandwidth estimation method.

### **I.3. Overview of the thesis**

The topics that we have introduced are discussed in detail in the following chapters. Each chapter opens with a very short outline of its content, followed by a brief introduction to the argument it treats and an overview of the related state of art. Afterwards, the dissertation continues by investigating the topic and presenting the original contributions that the thesis yields. The chapter closes with the summary of the problems that have been considered and the main contributions given.

The structure of the thesis is as follows. Chapter II outlines the classic mathematic characterization of fading effects in indoor environments and defines a possible Markov model for radio channels. Chapter III gives a general overview of the Bluetooth standard, lingering over the characteristics that are addressed in the rest of the thesis. Chapter IV deals with the performance evaluation of a Bluetooth point-to-point radio link and its mathematic modeling. Chapter V expounds the duality principle between master and gateway roles in Bluetooth scatternets and proposes a new scheduling scheme that aims to achieve fairness among users. Chapter VI describes a hybrid architecture of UMTS and Bluetooth and analyzes performance and fairness aspects. Chapter VII focuses on the analysis, modeling, and performance evaluation of the Contention-TDMA protocol for LMDS systems. Chapter VIII treats the problems concerning the transmission of TCP protocols over wireless links and discusses an analytic model for TCP Westwood. Finally, Chapter IX concludes the dissertation by summarizing the topics considered in the thesis and highlighting the original contributions given.

---

## II. INDOOR RADIO CHANNELS

---

*Radio data channels are generally characterized by high error probabilities and time-variant impulse responses that strongly impact on system performance. Analysis and simulation of a radio system require, therefore, an adequate description and modeling of the radio medium. This chapter deals with the typical characterization of fading effects in indoor environments and defines a possible Markov description for the radio channel.*

### II.1. Characterization of indoor radio channel

The channel impulse response of a wireless channel looks like a series of pulses because of multi-path reflections. Therefore, when a continuous waveform is transmitted, different copies of the transmitted signal may reach the receiving antenna with different time-delays and attenuation factors. These copies may constructively or destructively interfere at the receiver, producing a fluctuation of the received signal envelope known as *fading*. On the basis of the characteristics of this phenomenon, the radio channel can be characterized in four categories:

- flat (frequency-nonselective) slow fading channel models;
- flat (frequency-nonselective) fast fading channel models;
- frequency selective slow fading channel models;
- frequency selective fast fading channel models.

The fading is said *flat* when its amplitude can be assumed constant over all the signal bandwidth, otherwise is said *frequency selective*. Moreover, the fading is said *slow* when the amplitude value is varying very slowly with respect to the varying time of the signal envelope, while is said *fast* when its envelope changes significantly over a symbol period.

The following subsections offer a general description of a multi-path radio channel and derive the mathematical characterization of the fading phenomenon.

### II.1.A. Multi-path channels

A radio signal may follow many different propagation paths reach the receiver, each one associated with a time-variant propagation delay and attenuation factor. Consider the transmission of the generic band-pass signal

$$s(t) = \Re\{u(t)e^{j2\pi f_c t}\}, \quad (\text{II.1})$$

where  $u(t)$  is the complex low-pass signal,  $f_c$  is the carrier frequency, and  $\Re\{z\}$  denotes the real part of  $z$ . The received band-pass waveform is, then,

$$x(t) = \sum_n \alpha_n(t) s(t - \tau_n(t)) = \Re\{r(t)e^{j2\pi f_c t}\}, \quad (\text{II.2})$$

where the received complex low-pass signal  $r(t)$  is given by

$$r(t) = \sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} u(t - \tau_n(t)) \quad (\text{II.3})$$

and  $\alpha_n(t)$  and  $\tau_n(t)$  are the amplitude and time delay, respectively, associated with the  $n$ -th path. The equivalent low-pass channel is, thus, described by the time-variant impulse response

$$c(\tau; t) = \sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} \delta(t - \tau_n(t)). \quad (\text{II.4})$$

Considering the transmission of an unmodulated carrier at frequency  $f_c$ , the complex low-pass received signal could be expressed as,

$$r(t) = \sum_n \alpha_n(t) e^{-j\phi_n(t)}, \quad (\text{II.5})$$

with

$$\phi_n(t) = 2\pi f_c \tau_n(t). \quad (\text{II.6})$$

Thus, the received signal consists of the sum of a number of time-variant vectors with amplitudes  $\alpha_n(t)$  and phases  $\phi_n(t)$ , which generate the fading phenomenon. Note that, while the attenuation factors do not change drastically with small variations of sender and receiver position, the phases could change by  $2\pi$  radiant whenever  $\tau_n(t)$  changes by  $1/f_c$ . Therefore, since radio systems usually work at very high frequency, even relatively small motion of the medium could produce large changes of the phasors phase, and consequently of the received signal amplitude.

When the number of paths  $N$  is large, the impulse response  $c(\tau;t)$  is modeled as a complex Gaussian random process, i.e. its inphase and quadrature components are two independent Gaussian process [60],[76],[62]. In absence of Line-of-Sight (LOS) the envelope  $|c(\tau;t)|$  has a Rayleigh statistic. A Rayleigh-distributed random variable, with unit statistic power, is characterized by a probability density function (pdf) given by

$$f_a(a) = 2ae^{-a^2} 1(a), \quad (\text{II.7})$$

where  $1(a)$  is the step function.

When at the receiving antenna arrives also a strong direct component, i.e. in presence of LOS, the fading amplitude is well described by Ricean statistic and almost one of fading components has no-zero mean. For a Rice random variable with unit power, the probability density function is

$$f_a(a) = 2a(1+K) \exp[-K - a^2(1+K)] I_0(2a\sqrt{K(1+K)}) 1(a), \quad (\text{II.8})$$

where  $I_0(\cdot)$  is the zero-th order modified Bessel function of the first kind,

$$I_0(x) = \frac{1}{2\pi} \int_0^{2\pi} e^{-x\cos\vartheta} d\vartheta, \quad (\text{II.9})$$

and  $K$  is the Rice factor. The Rice factor is used to characterize the first path in the case of line-of-sight (LOS) between the transmitter and the receiver antennas. In this case, the electromagnetic field of the first ray is composed of a deterministic component and

a random (fluctuating) component. The ratio between the deterministic received signal power  $A^2$  and the power of the total fluctuating components  $M_m$  gives  $K$ , namely

$$K = \frac{A^2}{M_m}. \quad (\text{II.10})$$

By assuming that the power delay profile of the delay channel has been normalized to unit area, i.e.

$$A^2 + M_m = 1, \quad (\text{II.11})$$

we get the following relationship between  $K$  and  $A$ , namely

$$A = \sqrt{\frac{K}{K+1}}. \quad (\text{II.12})$$

Note that the case  $K = 0$  returns the Rayleigh pdf, while the case  $K = \infty$  models an additive white Gaussian noise channel (AWGN), with complete absence of fading. Typical reference values of  $K$  are 3 dB and 10 dB.

### II.1.B. Delay Spread

As previously introduced, the fading effect can be flat or frequency selective. This characterization depends on the relation between the signal bandwidth and delay spread of the radio channel impulse response. Indeed, if the impulse response is contained within a symbol period then inter-symbol interference (ISI) is limited. On the contrary, if the impulse response is time spread over a large interval, the interference produced could be varying in the signal bandwidth.

In order to characterize the delay introduced by the channel, we define the (local-mean) average power that is received with interval  $(T, T + dt)$ . Such characterization for all  $T$  gives the **delay profile**  $M(\tau)$  of the channel. The delay profile determines the frequency dispersion, that is, the extent to which the channel fading at two different frequencies  $f_1$  and  $f_2$  is correlated [60],[76],[62].

We define the *maximum delay time spread*,  $T_m$ , as the total time interval during which reflections with significant energy arrive. Moreover, we consider the *root mean square (rms) delay spread*  $T_{RMS}$ , defined as the standard deviation value of the delay of reflections, weighted proportional to the energy in the reflected waves:

$$T_{RMS}^2 = \frac{\int_{-\infty}^{+\infty} (\tau - \bar{\tau})^2 M(\tau) d\tau}{\int_{-\infty}^{+\infty} M(\tau) d\tau}, \quad (\text{II.13})$$

where

$$\bar{\tau} = \int_{-\infty}^{+\infty} \tau M(\tau) d\tau. \quad (\text{II.14})$$

If this value is roughly shorter than one tenth of symbol duration  $T$ , no serious ISI is likely to occur and then we could consider a flat fading channel model. In this case, the fading can be represented as a multiplication of the transmitted signal with the fading process, i.e. the channel has a time-variant impulse response given by

$$\begin{aligned} c(\tau; t) &= \left( \sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} \right) \delta(t - \tau), \\ &= a(t) e^{-j\varphi(t)} \delta(t - \tau), \\ &= \alpha(t) \delta(t - \tau); \end{aligned} \quad (\text{II.15})$$

with

$$a(t) = \left| \sum_n \alpha_n(t) e^{-j\phi_n(t)} \right|, \quad (\text{II.16})$$

$$\varphi(t) = \arg \left( \sum_n \alpha_n(t) e^{-j\phi_n(t)} \right), \quad (\text{II.17})$$

$$\alpha(t) = a(t) e^{-j\varphi(t)}. \quad (\text{II.18})$$

In indoor environments, the propagation paths length is generally on the order of few meters. Since a radio signal propagates at the light-speed, the maximum time-delay of a reflected ray is on the order of tens of nanoseconds, much shorter than the symbol period of typical high data-rate radio systems. Therefore, for the kind of applications we consider, the indoor radio channel can be considered a *flat fading radio channel*. The fading process can be, then, represented with a multiplicative complex function,  $a(t)$ , whose envelope is Rayleigh or Rice distributed for any  $t$ . Normalizing to one the average statistic power of the fade envelope, the signal to noise ratio (SNR) at the receiving side,  $\Gamma(t)$ , can be expressed as  $\Gamma(t) = u(t) \cdot \bar{\Gamma}$ , where  $\bar{\Gamma}$  is the expected value of  $\Gamma(t)$ , i.e.  $\bar{\Gamma} = E[\Gamma(t)]$ , while  $u(t) = |a(t)|^2$  is the square of the fading envelope. The pdf of  $\Gamma(t)$  is then given by:

$$f_{\Gamma}(\alpha) = \int_{-\sqrt{\alpha/\bar{\Gamma}}}^{+\sqrt{\alpha/\bar{\Gamma}}} f_a(\alpha) d\alpha, \quad \alpha \geq 0; \quad (\text{II.19})$$

where  $f_a(\alpha)$  is given by (II.7) or (II.8). In particular, in the case of Rayleigh fading, i.e. in absence of LOS between transmitted and receiver, the fading envelope squared,  $u(t)$ , has an exponential distribution and the pdf of  $\Gamma(t)$  becomes:

$$f_{\Gamma}(a) = \frac{1}{\bar{\Gamma}} \exp\left(-\frac{a}{\bar{\Gamma}}\right) 1(a). \quad (\text{II.20})$$

In the following of this work, we focus on flat fading radio channels only.

### II.1.C. Doppler Spectrum

One of the most critical aspects of data transmission over radio channel is the time-varying response of the transmission medium. Motion of an antenna, indeed, produces Doppler shifts in the frequency of the incoming waves. The frequency shift of the incident plane wave arriving at angle  $\theta$  is said *Doppler shift* and is given by

$$\Delta f_c = f_m \cos \theta. \quad (\text{II.21})$$

In (II.21),  $f_m$  is the maximum Doppler spread and is equal to

$$f_m = \frac{v_m}{\lambda_c} = \frac{v_m f_c}{c}, \quad (\text{II.22})$$

where  $v_m$  is the relative velocity between transmitter and receiver,  $f_c$  is the carrier frequency and  $c$  is the light velocity. If a sinusoidal signal (represented by a spectral line in the frequency domain) is transmitted, after transmission over a multi-path channel we will receive a power spectrum that is spread according to the power spectrum density (psd) function  $S(f)$ . The frequency range where the power spectrum is nonzero defines the Doppler spread, whereas the psd  $S(f)$  is said *Doppler Spectrum*.

The isotropic scattering model<sup>1</sup>, leads to the U-shaped power spectrum [60],[76],[62],

$$S(f) = \begin{cases} \frac{1}{2\pi f_m} \frac{1}{\sqrt{1 - \left(\frac{f}{f_m}\right)^2}} & |f| \leq f_m, \\ 0 & \text{otherwise.} \end{cases} \quad (\text{II.23})$$

Then, motion of the antenna leads to (time varying) phase shifts of individual reflected waves. Thus, their relative phases change all the time, varying the amplitude of the resulting composite signal.

Therefore, the Doppler spread  $f_m$  determines the varying speed of the channel, and

$$(\Delta t)_c \approx \frac{1}{f_m}, \quad (\text{II.24})$$

represents the coherence time of the channel. Clearly, a slowly changing channel has a large coherence time or, equivalently, a small Doppler spread. The memory of the fading components determines the classification of the radio channel in *slow* or *fast* fading channel. In particular, the fading process is said *fast* whether the coherence time

---

<sup>1</sup> The JTC channel model for indoor areas assumes a flat Doppler:  $S(f) = \frac{1}{2\pi f_m} \text{rect}\left(\frac{f}{2f_m}\right)$

is smaller than a symbol period  $T_b$ . On the contrary, i.e. whether the coherence time extends over many symbol periods, the fading is said *slow*. Note that even if the terminal remains fixed, moving obstacles around it (persons, doors, etc.) could produce Doppler spreading in the received signal, so the effects of such factors are included in the  $v_m$  value.

The Doppler spread is relevant, for instance to compute threshold crossing rate and average fades duration, as explained in the followings.

### II.1.D. Threshold Crossing Rate

Multi-path radio channel may generally result in rapid fluctuations of the received signal amplitude. If a certain minimum (threshold) signal level is needed for acceptable communication performance, the received signal will experience periods of:

- sufficient signal strength or "non-fade intervals", during which receiver can work reliably and at low bit error rate;
- insufficient signal strength or "fades", during which the bit error rate inevitably is close to one half (randomly guessing ones and zeros) and receiver may even fall out of lock. In this case the system experiments an outage period.

The average number of times per second a fading signal crosses a certain threshold is called threshold-crossing rate (TCR). To obtain the expression of the TCR we need some notations. Let  $a(t)$  be as defined in (II.16) and  $\dot{a}(t)$  be the derivative of  $a(t)$ . Furthermore, we define the indicator function of the event "envelope  $a(t)$  is in the interval  $[R, R + dr]$  with derivative  $\dot{a}(t)$ " as:

$$\chi(t) = \begin{cases} 1 & a(t) \in [R, R + dr], \dot{a}(t) = \dot{a} \\ 0 & \text{otherwise} \end{cases} \quad (\text{II.25})$$

If the crosses of threshold  $R$  with width  $dr$  occur with constant derivative  $\dot{a}$ , then each crosses lasts for  $dt = dr / \dot{a}$  seconds. Assuming that  $a(t)$  crosses the threshold

always with the same derivative  $\overset{\circ}{a}$  and integrating the indicator function  $\chi(t)$  over a time interval  $[0, T]$ , we obtain the following result:

$$\int_T \chi(u) du = n_T dt \quad (\text{II.26})$$

where  $n_T$  is the average number of threshold crossing with derivative  $\overset{\circ}{a}$ , during the interval  $[0, T]$ .

Denoting with  $f_{a, \overset{\circ}{a}}(a, \overset{\circ}{a})$  the joint probability distribution function of  $a$  and  $\overset{\circ}{a}$ , the average number of crossings per second, with derivative  $\overset{\circ}{a}$ , is equal to:

$$\begin{aligned} \lim_{T \rightarrow \infty} \frac{1}{T} n_T &= \frac{1}{T} \int_T \chi(u) du \frac{1}{dt} = E[\chi] \frac{1}{dt} = \\ &= \text{P} \left( a \in [R, R + dr], \overset{\circ}{a} \right) \frac{1}{dt} = f_{a, \overset{\circ}{a}} \left( R, \overset{\circ}{a} \right) \frac{dr}{dt} = f_{a, \overset{\circ}{a}} \left( R, \overset{\circ}{a} \right) \overset{\circ}{a}, \end{aligned} \quad (\text{II.27})$$

where in the third step we have assumed the ergodicity of the  $\chi(t)$  process. Finally, the expected crossing rate of the envelope level  $R$  with a positive slope is found by integrating over all possible derivatives:

$$L_R = \int_0^{\infty} f_{a, \overset{\circ}{a}} \left( a, \overset{\circ}{a} \right) \overset{\circ}{a} \cdot d \overset{\circ}{a} \quad (\text{II.28})$$

The joint pdf of signal amplitude and its derivative for Ricean fading can be found, for instance, in [76] that gives also the following expression for the average threshold crossing rate,

$$L_R = \sqrt{2\pi(K+1)} f_m \rho e^{-K-(K+1)\rho^2} I_0 \left( 2\rho \sqrt{K(K+1)} \right), \quad (\text{II.29})$$

where

$$\rho = \frac{R}{\sqrt{\mathbf{E}[a(t)^2]}}; \quad (\text{II.30})$$

is the so-called *fade margin*, while  $I_0(\cdot)$  is the zero order modified Bessel function of the first kind, defined in (II.9). For Rayleigh fading ( $K=0$ ), the (II.29) becomes

$$L_R = \sqrt{2\pi} f_m \rho e^{-\rho^2}. \quad (\text{II.31})$$

### II.1.E. Average Fade, Non-Fade period

Another quantity of interest is the *average fade period*, i.e. the average time that the envelope  $a$  of the received signal, remains below a specific level  $R$  (threshold).

Let  $t_i$  be the duration of the  $i$ -th fade period. Considering a very long time interval  $T$ , the probability that the envelope is less than  $R$  is

$$P_r[a \leq R] = \frac{1}{T} \sum_i t_i. \quad (\text{II.32})$$

The average fade duration is, thus, given by

$$AFD = \frac{1}{TL_R} \sum_i t_i = \frac{P_r[a \leq R]}{L_R}. \quad (\text{II.33})$$

If the envelope has Rice distribution we have

$$P_r[a \leq R] = \int_0^R f_a(u) du = 1 - Q\left(\sqrt{2K}, \sqrt{2(K+1)\rho^2}\right), \quad (\text{II.34})$$

where  $f_a(u)$  is the Ricean pdf and  $Q(a,b)$  is the Marcum Q function. Therefore,

$$AFD = \frac{1 - Q\left(\sqrt{2K}, \sqrt{2(K+1)\rho^2}\right)}{\sqrt{2\pi(K+1)} f_m \rho e^{-K-(K+1)\rho^2} I_0\left(2\rho\sqrt{K(K+1)}\right)}. \quad (\text{II.35})$$

If the envelope is Rayleigh distributed, the Eq.(II.32) and Eq.(II.33) become:

$$P_r[a \leq R] = \int_0^R f_a(u) du = 1 - e^{-\rho^2}; \quad (\text{II.36})$$

$$AFD = \frac{e^{\rho^2} - 1}{\sqrt{2\pi} f_m \rho}. \quad (\text{II.37})$$

The average non-fade duration (*ANFD*) is given by

$$ANFD = \frac{P_r[a > R]}{L_R} = \frac{1}{L_R} - AFD = T_F - AFD, \quad (\text{II.38})$$

where  $T_F$  is the average fade/non-fade cycle duration.

We define the *Outage Probability* as the probability that the channel will be in a fade period, i.e. the time percentage of channel fading. It is given by

$$P_{out} = P_r[a \leq R] = AFD \cdot L_R. \quad (\text{II.39})$$

**TAB. II.1. DECT AND BT FADING CHARACTERIZING PARAMETERS**

	<b>Frequency Band (<math>f_c</math>)</b>	<b>Carrier wavelength (<math>\lambda_c</math>)</b>	<b>Symbol period (<math>T_b</math>)</b>	<b>Slot period (<math>T_s</math>)</b>
DECT	1.8 GHz	16 cm	0.868 $\mu$ s	417 $\mu$ s
BT	2.4GHz	12 cm	1 $\mu$ s	625 $\mu$ s

### II.1.F. Fading process for Bluetooth and DECT radio systems

This subsection gives a graphical interpretation of the fading phenomenon in typical indoor environment. For this purpose, we consider two well known radio data protocols, namely the Digital Enhanced Cordless Telecommunications (DECT) and Bluetooth (BT) systems. The protocols have been both designed to work typically in indoor environments, but they are different for frequency band and symbol duration. Tab. II.1 summarizes the parameters, which the fading depends on, for the two standards.

In indoor environments, systems like DECT and BT are characterized by frequency non-selective fading because the multi-path spread of the channel,  $T_m$ , is short in comparison to the symbol period of the signal,  $T_b$  [62]. In fact, typical delay spread values of indoor DECT and BT transmissions are 108 and 162 ns [12] which are small with respect to the bit period of the two protocols, equal to  $T_b = 868$  ns and  $T_b = 1000$  ns, respectively. So far as the time variations of the radio channel is concerned, we recall that the coherence time, given by (II.24), determines the varying speed of the channel. DECT and BT wavelengths are approximately equal to  $\lambda_c \approx 16$  cm and  $\lambda_c \approx 12$  cm, respectively. Therefore, assuming that in indoor environments the equivalent relative speed between transmitter and receiver,  $v_m$ , can approximately be considered equal to 2 m/s [2], the minimum coherence time is approximately of 80 ms. Since the slot duration in DECT and BT are 0.417 ms and 0.625 ms, respectively, a very slow fading channel model could be used. Concluding, for DECT and BT systems, a flat slow fading channel model is considered.

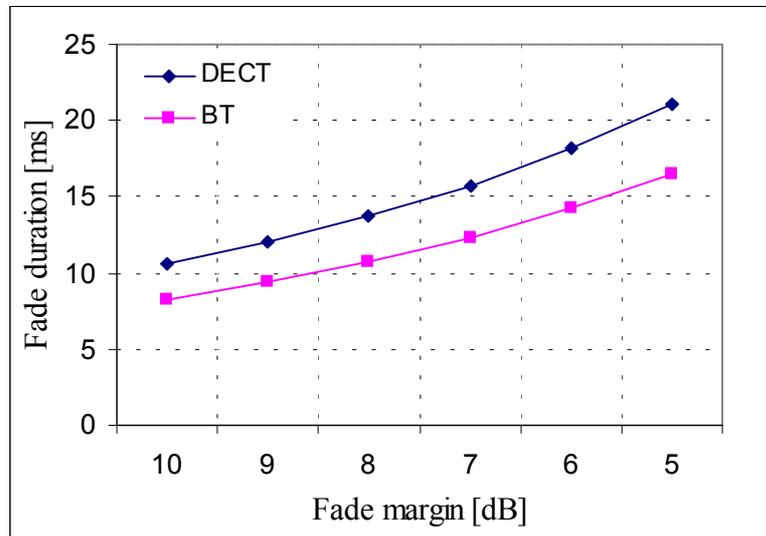


Fig. II.1 Average fade duration at BT and DECT frequencies ( $v_m=2$  m/s)

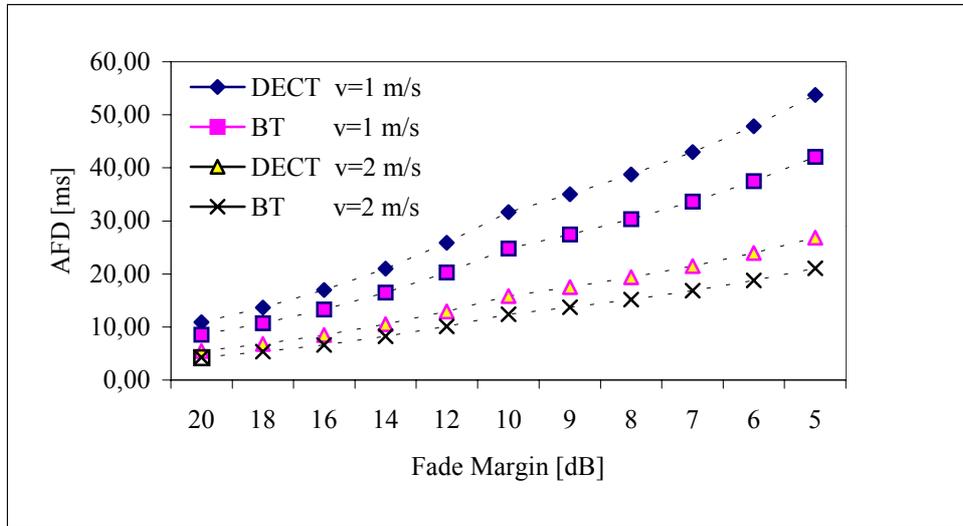


Fig. II.2 Rician AFD at DECT and BT frequencies,  $K=3$  dB

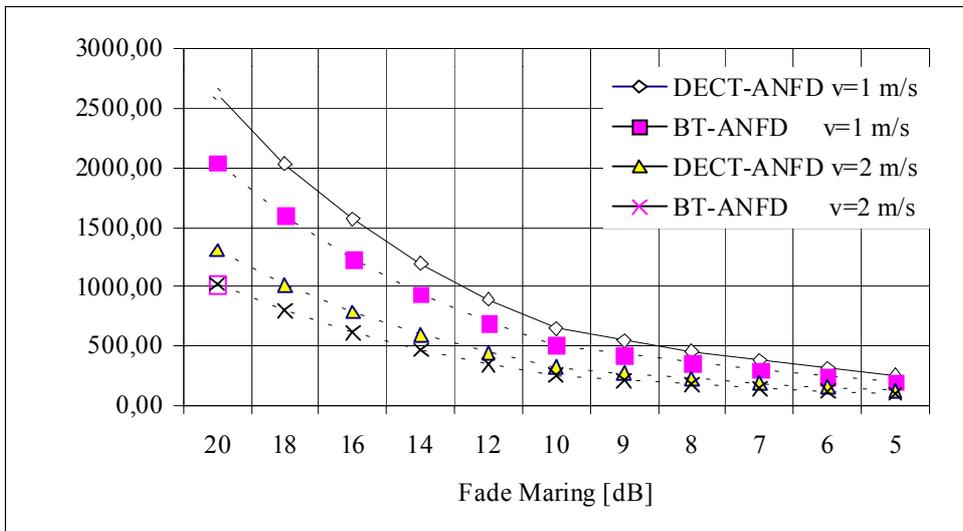


Fig. II.3 Rician ANFD at DECT and BT frequencies,  $K=3$  dB

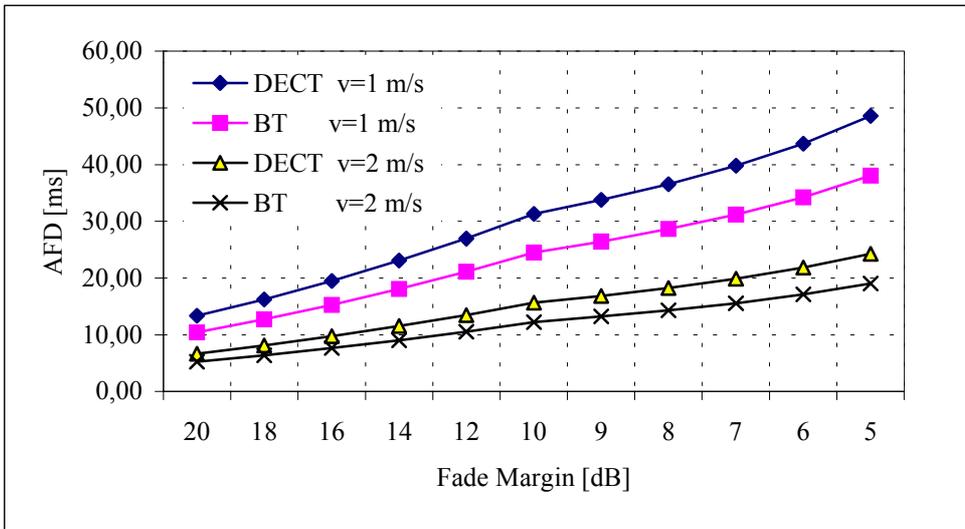


Fig. II.4 Rician AFD at DECT and BT frequencies, K=6 dB

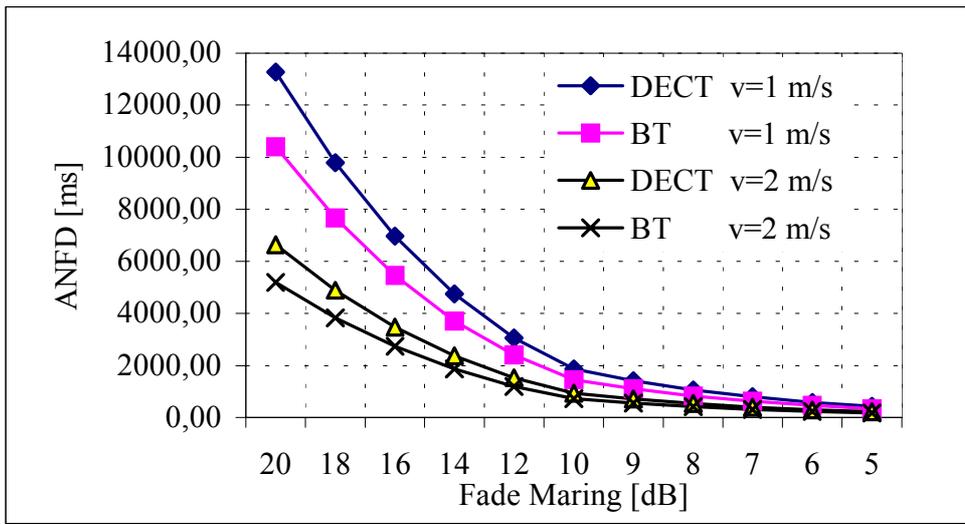
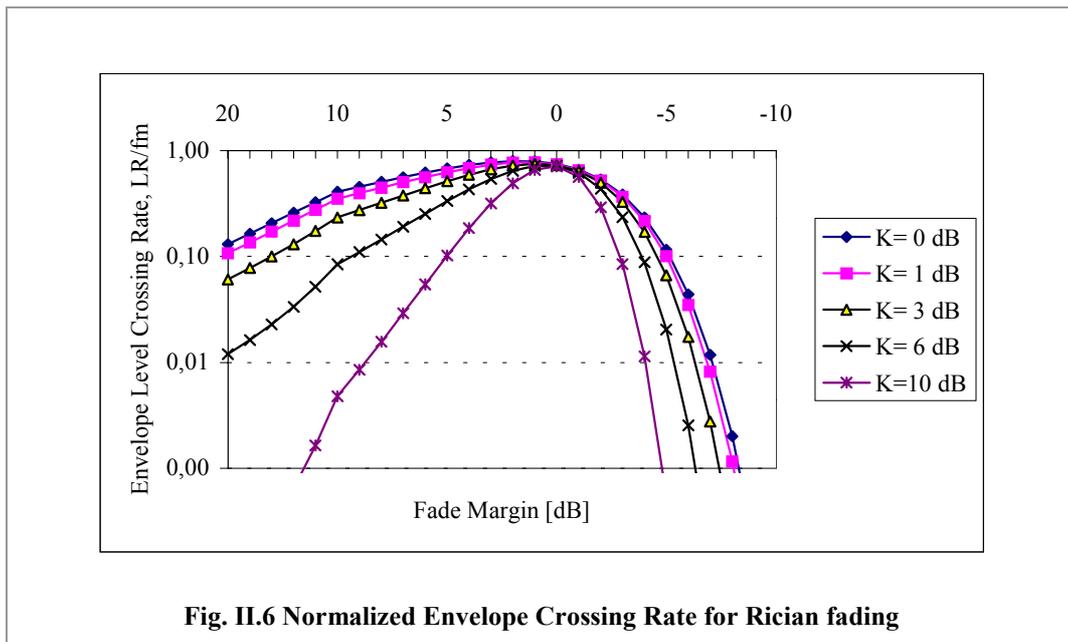


Fig. II.5 Rician ANFD at DECT and BT frequencies, K=6 dB

Fig. II.1 shows the average fade period duration of both DECT and BT systems, with Rayleigh fading, for decreasing values of the fade margin. Notice that, with fixed Doppler spread and fade margin, the average fade duration is shorter in BT than in DECT, even if the fade periods occur more frequently in the first case than in the second (the outage probability, in fact, must be the same since it is independent of the Doppler spread).

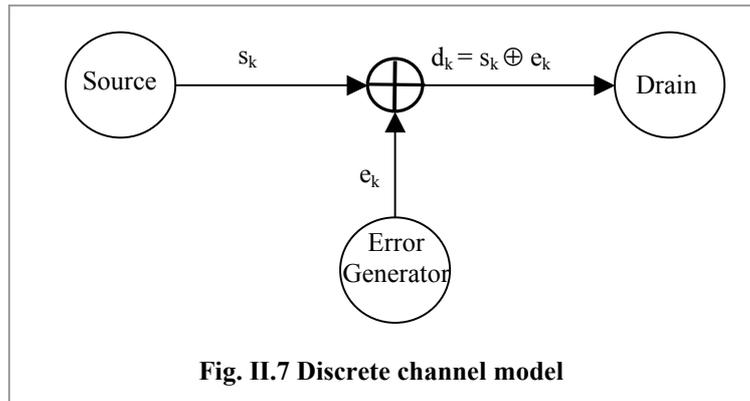
Fig. II.2–Fig. II.5 show the average fade and non–fade periods duration at the DECT and BT operating frequencies, for many different values of fade margin and for two rice factors,  $K = 3$  dB,  $K = 6$  dB. We can note that the fade and non–fade periods are strongly affected by the equivalent relative speed between transmitter and receiver ( $v_m$ ): higher is the relative speed, shorter are, on average, the fade and non–fade periods, i.e. the fade cycle duration. Note that the outage probability depends only on the fade margin and the Rice factor; the Doppler effect determines only the average duration of the fade period but does not affect the overall probability that the signal is in such a period. Instead, the outage probability depends strongly on the Rice factor. We can notice that, while the average fade period duration is only marginally influenced by the Rice factor, it impacts strongly on the average non–fade period duration. A higher Rice factor, in fact, determines prolongation of the non–fade periods, reducing the overall outage probability.



This observation is further confirmed by the curves depicted in Fig. II.6, which show the envelope level crossing rate, i.e. the frequency with which the normalized signal envelope crosses a given threshold, for different values of Rice factor and fade margin. We can note that, for higher values of the Rice factor, the average duration of a fade cycle is normally longer, as pointed out comparing the previous graphs.

## II.2. Discrete channel models for digital radio links

Discrete channel models for digital radio links have been extensively investigated by the scientific community, since they are simple and suitable to be implemented in computer simulation architectures. A digital channel model is a simplification of the transmission scheme in which the different blocks of the real transmission chain are collapsed into three macro-blocks with binary inputs and outputs, as represented in Fig. II.7. The macro blocks represent the *Source* of the bit stream (including source coder and channel coder), *Error generator* and *Drain*. The output  $\{e_i\}_{i \in N}$  of the Error generator block gives the error process at the receiving side, so that 0's and 1's in the output stream of this block denote good and erroneous bit receptions, respectively.



Several models related to finite-state Markov chains (FSMC) have been proposed for the generation of such error processes in the search for a good trade-off between model simplicity and results accuracy. These models are based on the general idea of associating each state of the Markov chain (MC) with a binary symmetric channel (BSC), with a given crossover probability. The crossover probabilities are chosen in order to identify a specific channel quality. One of the earlier works in this direction is due to Gilbert [32], who investigated a Markovian model with a *Good* state (G), assumed error-free, and a *Bad* state (B) in which errors could occur with a given probability. In [24], Elliot extended the model by assuming a low bit error probability even in state G. Successively, Fritchman [28] generalized the model by considering a Markov chain with  $N$  states partitioned in two groups: error-free states and error states. Transitions among states occurred synchronously with the transmission of the source symbols, producing the binary error sequence. A comparison among these three models

for a GSM-like transmission system can be found in [77], in which the authors summarize the main characteristics of each model and successively present the simulation results for a GMSK-based system in a Rayleigh fading environment.

An interesting theoretical approach to the use of FSMC for modeling radio communications channel can be found in [84], where authors describe a simplified method to derive the statistical parameters of a Markov model in the case of Rayleigh fading channel. This general model allows deriving Gilbert and Elliot models as particular cases. For this reasons it is briefly summarized in the next subsection.

### II.2.A. A general Markovian model for fading channel

A mathematical model for the radio link, based on Markov chain (MC), may be constructed by partitioning the range of the received SNR into a finite number of intervals [84]. Given a sequence of threshold values,  $\{\Lambda_k\}$ , so that  $0 = \Lambda_0 < \Lambda_1 < \Lambda_2 < \dots < \Lambda_{N_\Sigma} = \infty$ , the fading channel is said to be in state  $s_k$ ,  $k = 0, 1, 2, \dots, N_\Sigma - 1$ , if the received SNR is in the interval  $[\Lambda_k, \Lambda_{k+1})$ . Each state of the channel is associated with a binary symmetric channel with a given crossover probability. Assuming a first-order Markovian model, the state evolution depends only on the current state, on the basis of an appropriate state transition matrix  $\mathbf{T}$ , while any other previous state is independent of the current one. This assumption is generally accepted since it has been proven that, given the information corresponding to the previous symbol, the amount of uncertainty remaining in the current symbol is negligible [85]. Then, the model may be described as a 4-ple,  $\Omega = (T_s, \Sigma, \mathbf{T}, \mathbf{e})$ , where:

- $T_s$  is the state transition step;
- $\Sigma = \{s_k\}_{k=0,1,\dots,N_\Sigma-1}$  is the set of possible states, with cardinality  $N_\Sigma$ ;
- $\mathbf{T}$  is a  $N_\Sigma \times N_\Sigma$  state transition probability matrix such that  $t_{K,Y} = [\mathbf{T}]_{(K,Y)}$  represents the transition probability from state  $K$  to state  $Y$ , with  $K, Y \in \Sigma$ ;
- $\mathbf{e}$  is an  $N_\Sigma \times 1$  error probability vector so that  $e_K = [\mathbf{e}]_{(K)}$  represents the crossover probability of the binary symmetric channel associated to the state  $K \in \Sigma$ .

As explained in [84], the steady state probability for the generic state  $s_k$  can be computed by integrating the SNR pdf over the relative SNR region:

$$P_k = \int_{\Lambda_k}^{\Lambda_{k+1}} f_{\Gamma}(\lambda) d\lambda. \quad (\text{II.40})$$

Notice that, in the simpler case of Rayleigh fading, the (II.40) can be explicated as follow:

$$P_k = \int_{\Lambda_k}^{\Lambda_{k+1}} f_{\Gamma}(\lambda) d\lambda = e^{-\frac{\Lambda_k}{\Gamma}} - e^{-\frac{\Lambda_{k+1}}{\Gamma}}. \quad (\text{II.41})$$

The state transition probabilities can be approximated by the following expressions:

$$t_{k,k+1} \approx \frac{N_{\Lambda_{k+1}}}{P_k/T_S} = \frac{N_{\Lambda_{k+1}}}{P_k} T_S, \quad k = 0, 1, \dots, N_{\Sigma} - 2, \quad (\text{II.42})$$

$$t_{k,k-1} \approx \frac{N_{\Lambda_k}}{P_k/T_S} = \frac{N_{\Lambda_k}}{P_k} T_S, \quad k = 1, 2, \dots, N_{\Sigma} - 1, \quad (\text{II.43})$$

where  $N_{\Lambda_k}$  is the threshold crossing rate associated to  $\Lambda_k$ . The remaining transition probabilities can be easily computed from these, since the sum of each row of the transition probability matrix must be one.

Finally, the crossover probability for the  $k$ -th state can be computed as:

$$e_k = \frac{\int_{\Lambda_k}^{\Lambda_{k+1}} f_{\Gamma}(\lambda) \varepsilon(\lambda) d\lambda}{\int_{\Lambda_k}^{\Lambda_{k+1}} f_{\Gamma}(\lambda) d\lambda} = \frac{\int_{\Lambda_k}^{\Lambda_{k+1}} f_{\Gamma}(\lambda) \varepsilon(\lambda) d\lambda}{P_k}, \quad k = 1, 2, \dots, N_{\Sigma} - 1, \quad (\text{II.44})$$

in which  $\varepsilon(\lambda)$  is the error probability as a function of the received SNR.

---

## III. BLUETOOTH RADIO SYSTEM

---

*Bluetooth is a short range low power radio technology, operating in the license-free ISM frequency band. Even though Bluetooth was initially proposed as cable replacement for connecting small electronic devices, progressively it is progressively gathering the attention of the major operators in the field of personal communications. This chapter gives a general overview of the standard and the rationale behind some technical choices. In particular, the chapter dwells upon the aspects that will be focused in the rest of the thesis.*

### III.1. Bluetooth: connecting without cables

Today, a large number of people utilize many different electronic devices (laptops, palmtops, cellular phones, cameras, etc) on the job as well as in the ordinary life. For instance, most mobile professionals travel with cell-phone and laptop, and discover that they often are using the phone to browse the web, or need to send e-mail out of their laptop to the network. The ability to interconnect phone and laptop instantly and unobtrusively without cables would make life much easier. This is the first type of application – cable replacement – that motivated the development of the Bluetooth wireless PAN, an extremely versatile, low cost network interface soon to be installed in all personal devices. From the initial cable replacement concept the range of Bluetooth applications has quickly expanded to more challenging scenarios including: ad hoc networking among several individuals (each equipped with a Bluetooth piconet); access to the Internet (via repeaters on the wall), and pervasive, “unconscious computing” (interaction between mobile user and the environment, e.g., parking meters and airport check in counters).

This chapter gives a brief overview of the Bluetooth technology, focusing on the features of primary interest for the purposes of this work. A complete description of such topics can be found, for instance, in [70] and [81].

### III.2. Radio spectrum and modulation

The Bluetooth system operates in the Industrial–Scientific–Medical (ISM) frequency band, available in most countries of the world from 2400 MHz to 2483.5 MHz. This spectrum, formerly reserved for some professional user groups, has recently been opened worldwide<sup>2</sup> for commercial use without the need for licenses. These characteristics are ideal for Bluetooth that aims at being economic and worldwide usable.

However, many others electronic devices take advantage of such a frequency band. As a matter of fact, the number of electronic devices operating in the ISM band, such as baby monitors, garage door openers, cordless phones and microwave ovens, has been drastically increasing in the last period. Unfortunately, these devices are potentially sources of mutual interference. Several different regulations have been issued by the various countries to limit the interference in the ISM band and to guarantee a fair utilization of the radio resource. According to these regulations, Bluetooth standard provides for a frequency–hopping spread spectrum (FHSS) modulation scheme and for three classes of equipment with different power capabilities. The Bluetooth channel is represented by a pseudo–random hopping sequence spanning 79 radio frequency (RF) channels, 1–MHz wide, uniformly distributed in the 80 MHz of the ISM band. The time is divided into consecutive time slots, where each slot corresponds to a RF hop frequency. The nominal hop rate is 1600 hops/s, corresponding to slot duration of  $T_{slot} = 625\mu s$ . This kind of modulation better supports low–cost, low–power radio implementations, since it requires narrowband electronic components. In addition, they better cope with near–far problems: a nearby jammer is effectively suppressed by the narrow channel filter as long as its jammer TX spectrum does not coincide with the selected hop channel. Clearly, whether interference jams a hop channel, it likely produces an erroneous reception of the radio packet, which has to be retransmitted in a different frequency band.

The power classes defined by the standard are:

---

<sup>2</sup> In reality, some country still have a different ISM band, but harmonization efforts are ongoing to smooth away these differences.

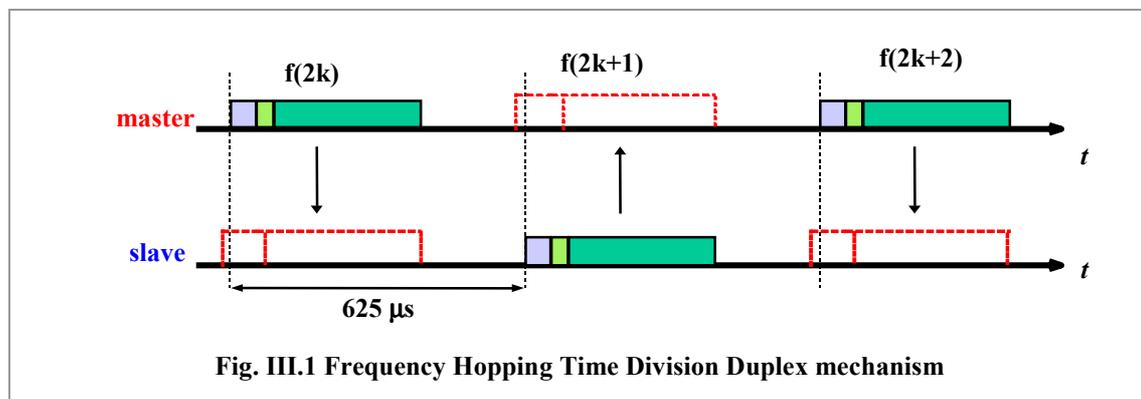
- *Class 1.* Outputs 100mW (+20 dBm) for maximum range. In this class, power control is mandatory ranging from 4 to 20 dBm. This mode provides the greatest distance: about 100m in indoor environment.
- *Class 2.* Outputs 2.5mW (+4 dBm) at maximum. Minimum power 0.25mW. Power control can be implemented, but is not necessary.
- *Class 3.* Lowest power. Nominal output is 1mW (0 dBm), covering a distance of roughly 10m in indoor environment.

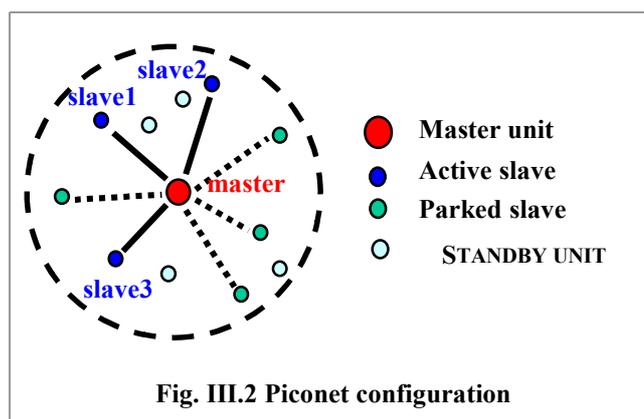
Power control is only required in Class 1 devices to keep the devices from emitting any more than the necessary RF power. In the case of Class 2 and Class 3 devices, power control is optional, but may be useful when implemented in low-power applications.

Data symbols are sent using a simple and robust binary modulation scheme, namely a Gaussian-shaped frequency shift keying (GFSK) modulation with a nominal modulation index of  $h$  between 0.28 and 0.35. Logical ones are sent as positive frequency deviations, logical zeros as negative frequency deviations, with a symbol period of  $1\mu s$ . Demodulation can simply be accomplished by a limiting frequency-modulation (FM) discriminator. This modulation scheme allows the implementation of low cost radio units.

### III.3. Channel Definition

A Bluetooth channel is identified by the frequency hopping sequence that determines the order which the different carrier frequencies have to be visited with. As already introduced, the channel is divided into consecutive slots, lasting  $625\mu s$  each. The full duplex connectivity is provided by means of a time-division duplex (TDD) scheme in





which subsequent slots are alternately used for transmitting and receiving, as shown in Fig. III.1. Separation of transmission and reception in time effectively prevents cross-talk between transmit and receive operations in the radio transceiver, which is essential if a one-chip implementation is desired.

### III.4. Bluetooth Piconets

Networking in Bluetooth is based on a small network structure, called *piconet*. In each Piconet, one unit acts as *master*, controlling the channel access in order to avoid collision and scheduling the traffic on the Piconet. All other units participating in the Piconet are *slaves* (Fig. III.2). Usually, the role of master is given to the unit that has started the connection, creating the piconet, while the units that accept the connection request get the role of slaves. However, these roles are tightly correlated to the piconet existence: when a unit leaves the piconet, its master or slave role is deleted.

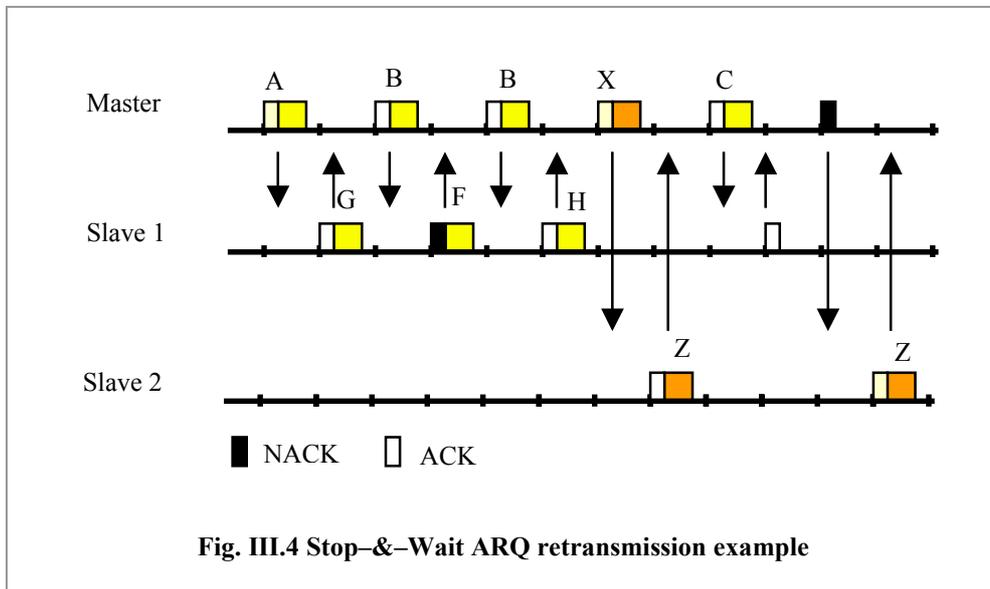
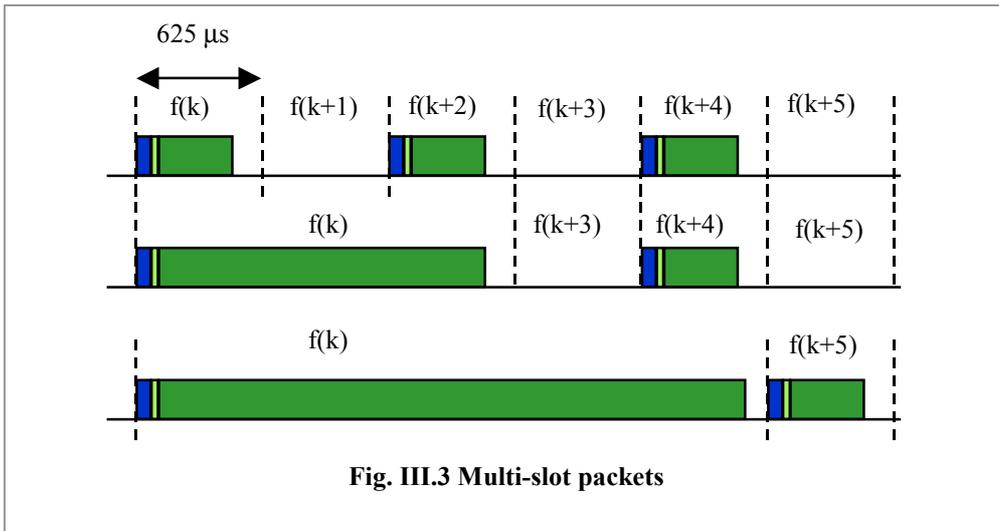
Each piconet is associated with a given FH/TDD channel, whose capacity of approximately 1Mbps is shared by all the active units in the piconet. The frequency hopping sequence that defines the channel is uniquely derived from the Bluetooth physical address of the master unit, while the clock value of this unit determines the phase in the hopping sequence. The piconet is, therefore, uniquely identified by the master unit parameters, and all the other units participating to the piconet have to know these parameters in order to synchronize on the same FH/TDD channel.

Different piconets have different masters and therefore also different hopping sequences and phases. This allows many piconets to share the same physical space without increasing excessively the mutual interference. However, since the frequency

hopping sequences are not orthogonal and the channels are asynchronous, interference among different piconets may sporadically occur. Thus, the theoretical capacity of 79 Mb/s cannot be reached, but is at least much larger than 1 Mb/s.

In a piconet, transmissions can directly occur between master and slaves only. Master-to-slave transmissions start on odd slots, while slave-to-master transmission on even slots. In order to prevent collisions, the master applies a polling technique for which a slave is allowed (and required) to transmit in a slave-to-master slot only if addressed by a master transmission in the preceding master-to-slave slots. If the master sends a data packet to a slave, the slave gets implicitly polled and can return information. If the master has no information to send, it may poll the slave explicitly with a short control packet (POLL). The slave addressed by a master transmission is required to reply immediately in order to acknowledge the reception of the master's packet. For this purpose, the slave can transmit a data packet (if any) or a special control packet (NULL). Since the master schedules the traffic in both the uplink and downlink, intelligent scheduling algorithms may be used to enhance the system performance.

A slave can be in two different states: *active* or *parked* (Fig. III.2). Each active slave is associated to a 3-bits temporary Active Member Address (AMA). The all-zeros address is assigned to the master and is used for broadcasting messages to all the active members in the piconet. The maximum number of active slaves in a piconet is, then, limited to eight. A much greater number of units can be, instead, in the parking state. Each parked slave is distinguished by an 8-bits Parked Member Address (PMA), which allows having up to 256 parked nodes in a piconet. Parked nodes are in a power-saving mode since they do not participate to the piconet activity. However, they keep the synchronization with the piconet holding the physical address and the clock offset of the master unit. Furthermore, parked units listen periodically to a common beacon channel to get a possible "*waking up*" message from the master and to keep the channel synchronization.



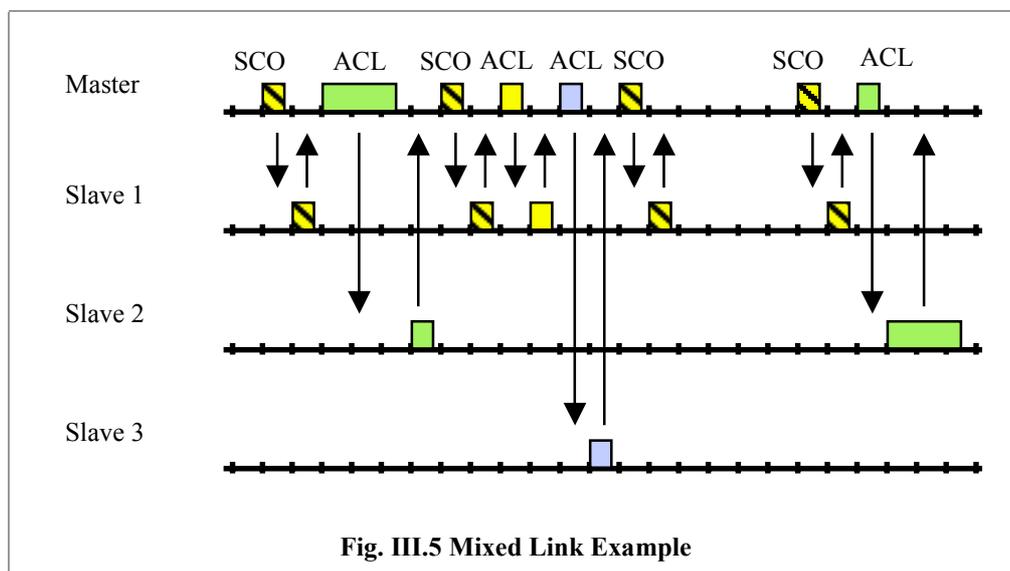
### III.5. Physical link definition

Two types of links have been defined to support voice and data:

- synchronous connection-oriented (SCO) link;
- asynchronous connectionless (ACL) link.

SCO links support symmetrical, circuit-switched, point-to-point connections typically used for voice. These links are defined on the channel by reserving two consecutive slots (forward and return slots) with a fixed period. Reservation is carried out by the master and slave when the link is set up. Voice packets are single-slot only and are never retransmitted, but they can be optionally protected by a 1/3 or a 2/3 FEC code.

ACL links support symmetrical or asymmetrical, packet-switched, point-to-multipoint connections, typically used for bursty data transmission. ACL links can use 1-slot, 3-slot, and 5-slot long data packets in order to increase the system capacity. When a multi-slot packet is used, the transmitter frequency remains unchanged for the whole packet duration, thus reducing the efficiency loss due to the phase-locked loop (PLL) settling time ( $\sim 220\mu s$ ), occurring each time a new frequency is used. After the multi-slot packet, the channel continues on the hop channel as dictated by the master clock (Fig. III.3).



Payload of data packets is covered by a 16-bit cyclic redundancy check (CRC) code, used by a simple Stop-and-Wait ARQ scheme to verify the integrity of received data. After reception and CRC control, a base-band packet is immediately acknowledged by using a 1-bit field in the header of the reply packet. In case of negative acknowledgement, or absence of valid response from the receiving side, the packet shall be retransmitted. In order to reduce the number of retransmissions, data packets can optionally be protected by a forward-error-correction (FEC) code, namely a (15,10) shortened Hamming code, with an efficiency of two-thirds. Each block of 10 information bits is, thus, encoded into a 15-bit codeword that is obtained by the generator polynomial  $g(D) = (D+1)(D^4 + D+1)$ . The code can correct all single errors and detect all double errors in each codeword.

The ACL link is constantly present between the master and the slave as long as the piconet exists, and the master can exchange ACL packets with any slave in the slots not reserved for SCO links, on a per-slot basis. A slave is permitted to return an ACL packet in the slave-to-master slot if and only if it has been addressed in the preceding master-to-slave slot, following the polling scheme previously mentioned. Fig. III.5 depicts mixed SCO and ACL links on a piconet with one master and three slaves. Slave 1 supports an ACL link and a SCO link with a six-slot SCO period. Slave 2 and 3 support ACL links only. Note that slots may be empty when no data is available.

TAB. III.1 CHARACTERISTICS OF THE SIX ACL PACKET FORMATS PROVIDED BY BLUETOOTH.

Pck Type		DM1	DH1	DM3	DH3	DM5	DH5
Number of Slot		1	1	3	3	5	5
Payl Head (bit)		8	8	16	16	16	16
Payl Data (bit)		136	216	968	1464	1792	2712
FEC		2/3	NO	2/3	NO	2/3	NO
Maximum bit-rate (Kbit/s)	Asymmetric Connection	108,8	172,8	387,2	585,6	477,8	723,2
		108,8	172,8	54,4	86,4	36,3	57,6
	Symmetric Connection	108,8	172,8	258,1	390,4	286,7	433,9
Tot Pck Size (bits)		366	366	1626	1622	2871	2870

Tab. III.1 summarizes the characteristics of different ACL packets. The notation “DMn” stands for “Medium Data-rate” and is used to indicate FEC-encoded data packets, while “DHn” stands for “High Data-rate” and denotes unprotected data packets. The number “n” following the acronyms is the length of the packet in slots.

### III.6. Packet-based communications

Communication in a piconet is packet-based. Each packet contains three main fields, as shown in Fig. III.6: the access code (AC), the packet header (HEAD) and, optionally, the payload field (PAYL).

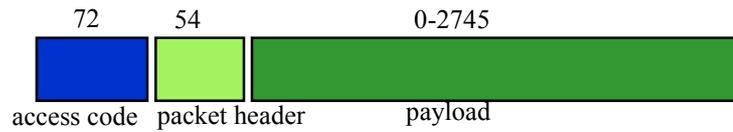


Fig. III.6 Packet format

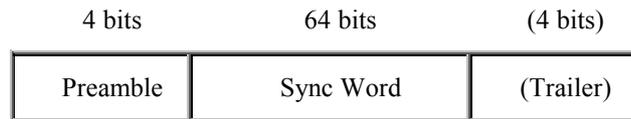


Fig. III.7 Access Code Field

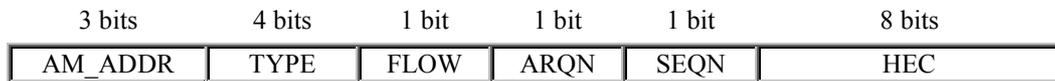


Fig. III.8 Packet Header field

### ***Access Code Field***

The 72-bit AC field is used for synchronization, DC offset compensation and identification. The access code field is organized as shown in Fig. III.7.

Beside a 4-bit preamble and a 4-bit trailer, the channel AC contains a 64-bit synchronization word, derived from the lower address part (LAP) of the master unit, i.e. the last 24-bit of the master physical address. There are 3 types of access codes:

- *Channel Access Code (CAC)*: identifies a piconet and is included in all packets exchanged on the piconet channel;
- *Device Access Code (DAC)*: used for special signaling procedures;
- *Inquiry Access Code (IAC)*: used for inquiry procedures.

DAC and IAC do not include the trailer bits and are 68 bits long. Every packet sent in the same piconet is preceded by the same CAC. A sliding correlator in the receiver of the Bluetooth unit correlates the incoming signal against the expected sync word, and triggers when a threshold is exceeded. This trigger signal is used to determine the receive timing. If the correlator output does not exceed the threshold, the received

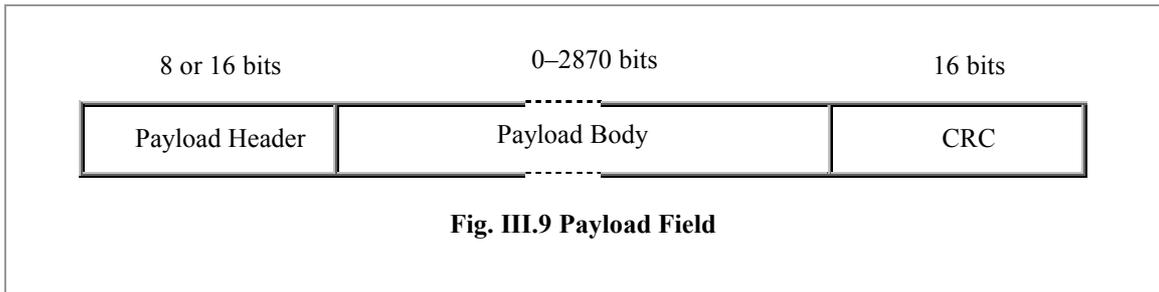
packet is not considered valid and the rest of its content is ignored. The construction of the sync words are based on a (64,30) expurgated block code, which guarantees a minimum Hamming distance of 14 between sync words based on different LAPs.

### ***Packet Header***

The packet header field (HEAD), which trails the access code, is structured as shown in Fig. III.8. The different parts have the following meaning:

- AM\_ADDR: 3-bit active member address.
- TYPE: 16 different types of packet can be distinguished. The TYPE code gives information on the physical link type associated with the packet (e.g. synchronous or asynchronous link) and simultaneously on the slot occupancy of the packet.
- FLOW: 1 bit used for flow control (only over asynchronous link):
  - FLOW=0 means RX buffer full, STOP transmission;
  - FLOW=1 means RX buffer empty, GO transmission;
- ARQN: 1-bit acknowledgment indication:
  - ARQN=1 means successfully packet reception;
  - ARQN=0 means unsuccessfully packet reception (Negative ACK)
- SEQN: 1-bit sequence number.
- HEC: 8-bit header-error-check.

The HEAD field contains important link control information, such as the active member address of the receiver, the type of the packet, and the bits used by the simple stop-&-wait automatic repeat request (ARQ) scheme that provides data transmission reliability as previously discussed. Furthermore, an 8-bit header error check (HEC) word completes the field. The total header, including the HEC, consists of 18 bits encoded with a 1/3 forward error correction (FEC) scheme, obtained by replicating each bit for three times, resulting in a 54-bit field. After the reception of a packet with valid AC field, Bluetooth receivers decode the HEAD field. If the HEC check fails or the AMA is different by its own, the reception is immediately stopped and the remaining part of the packet is ignored.



### ***Payload field***

Payload, which may or may not trail the header, consists of the following three parts pictured in Fig. III.9:

- Payload Header: one byte long for single-slot packets, two bytes long for multi-slot packets. The field contains, among other information, a length indicator for the number of bytes in the payload body only;
- Payload Body: contains the upper layer packet data unit (PDU) and then determines the effective throughput;
- Payload CRC: 16-bit CRC code, generated by the CRC-CCITT polynomial  $214004_8$ .

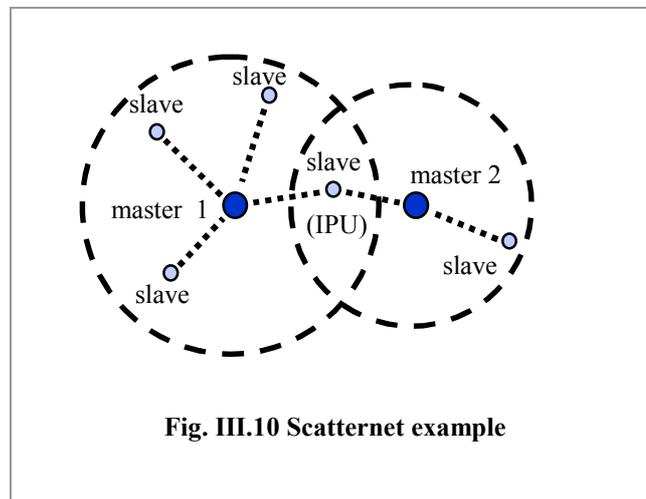
The length of the payload may vary from 0 to 2745 bits and it can be optionally protected by FEC codes.

## **III.7. Inter-Piconet Communications**

The Bluetooth system has been optimized to have tens of piconets operating in the same area without noticeable performance degradation. Due to the fact that Bluetooth uses packet-based communication over slotted links, it is possible to interconnect different piconets. This means that units can participate in different piconets even if, at any instant time, a unit can communicate in one piconet only, since the radio can tune to a single hop carrier at time. However, the unit can jump from one piconet to another by adjusting the piconet channel parameters (i.e., the master identity and clock). These units may act as *gateways*, allowing the exchange of data among the piconets they belong to. Indeed, the hop selection mechanism has been designed to allow inter-

piconet communications: by changing the identity and clock input to the selection mechanism, instantaneously a new hop for the new piconet is selected.

When two or more piconets in the same area are interconnected in such a way, the resulting wireless network is referred to as *scatternet* (Fig. III.10). While the Bluetooth standard defines gateway nodes, the actual mechanism and algorithms that this can be accomplished with is left open. Furthermore, traffic scheduling and routing in a scatternet with inter-piconet communications is a challenge and still a research topic.



**Fig. III.10 Scatternet example**

---

## IV. BLUETOOTH RADIO LINK: PERFORMANCE ANALYSIS & MODELING

---

*In this chapter we evaluate the performance of a point-to-point Bluetooth connection in different environmental conditions. To achieve this goal we carry out a large series of laboratory experiments, consisting of heavy data file transfers between a fixed file transfer protocol (FTP) server and a nomadic client, through a single-hop Bluetooth link. Several trials are conducted with different Bluetooth radio packet formats, in order to investigate possible performance tradeoff to varying of environmental conditions. Successively, experimental results are used to define a discrete channel model based on a three-state Markov chain, which is able to reproduce closely the radio link behavior for all the packet formats supplied by Bluetooth.*

### IV.1. Aims of the study

Bluetooth standard defines six packet formats for asynchronous data traffic (ACL), characterized by different values of payload capacity, error protection and time duration. Unprotected and long packet formats show high payload capacity but are sensitive to errors. On the contrary, short and protected formats are less subject to payload errors to the detriment of a lower capacity. Therefore, the performance yielded by such different packet formats may show a tradeoff at varying of radio channel conditions.

This chapter addresses this issue and gets a deep insight of Bluetooth radio link behavior in real-world situations. Such analysis may result useful, for instance, to develop efficient scheduling algorithms at MAC layer, that keep into consideration the actual performance that a given packet format yields rather than the nominal one. For such a purpose, we perform a series of experimental measurements, consisting on data transmissions over Bluetooth point-to-point radio link.

Furthermore, the investigation aims to define a simple mathematical channel model that would be able to approximately reproduce the radio channel behavior. The radio

models that are usually adopted to simulate the Bluetooth radio link are either very simple Gilbert models or complicate signal propagation models. In the former case, the error process at the receiving side is reproduced through a two–state Markov chain, in which a state represents good bit reception (no error), while the other state represents bad bit reception (non–zero error probability). Even though such a model has proven to be able to reproduce the error process with good accuracy in many cases, it does not appear suitable to capture the different characteristic of all the six packet formats supplied by Bluetooth. On the other hand, sophisticated channel models consider the aspects at physical layer of the connection, like modulation, signal propagation, multipath effect, in detail. Thus, these models may result unsuitable to be used in some simulation platforms because of their high complexity and the considerable computational resources that they require.

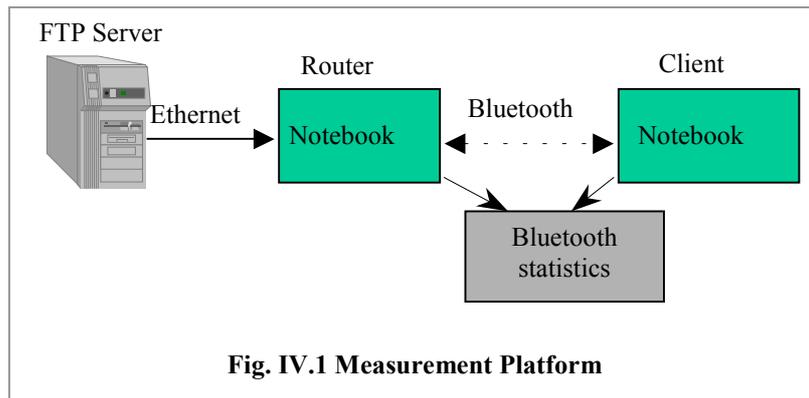
The study here presented leads to a mathematic model for the Bluetooth link that, against a very high accuracy of the results yielded, results almost as simple as Gilbert model.

## **IV.2. Experimental analysis of Bluetooth radio link**

We perform a series of large bulk data transfers (roughly 1 Mbytes) between a fixed file transport protocol (FTP) server and a nomadic FTP client. Experiments are performed in different real–world situations, obtained by moving the nomadic station around the research laboratory. However, experiments are carried out with no other wireless system working in ISM band active.

### **IV.2.A. Measurement platform and tools**

The system architecture used for collecting measurement is depicted in Fig. IV.1. The FTP server is connected to a router through a 10Base–T Ethernet. The router connects fixed and mobile radio parts establishing a Bluetooth piconet with the FTP client. The piconet was configured with the *master* on the router and the *slave* on the client to maximize the link capacity in the forward direction. Router and FTP client are run on two Pentium II notebooks, clocked to 200 MHz and using Windows 98 operating system. The radio interface runs a Bluetooth DigiAnswer firmware, release 4.02, where



Bluetooth base-band processing takes place. Winsock2 TCP/IPv4 Windows implementation is used in the tests. The DigiAnswer Bluetooth Application Programmer's Interface collects statistics regarding active Bluetooth link in terms of AC, FEC and CRC errors.

Unfortunately, the measurement tools available for this kind of experiments have some limitations, so that monitoring the radio link behavior suffers from some drawbacks. The probing program works independently on the master and slave device, thus the correlation of master transmissions to slave receptions, and vice versa, may be affected by little misalignment. Furthermore, the probing time is not always constant and may sporadically assume very high values. Therefore, we are able to determine the trend only of the radio connection, and not its step-by-step history. Nevertheless, the information collected permits some interesting insight into the Bluetooth performance.

Many tests are carried out in every environment, using different Bluetooth packet types. More precisely, only the packet format in the forward direction (from master to slave) is changed while, in the reverse direction, the single-slot unprotected packet (DH1) is always used since it is ideal to carry the low traffic in this direction mainly due to TCP acknowledgements.

#### IV.2.B. Target metrics

Due to the native retransmission mechanism provided by Bluetooth for ACL packets, the effective performance perceived by the upper layer protocols depends on the behavior of both forward (master to slave) and reverse (slave to master) connections. Though the radio channel is unique, transmissions in the two directions can show very

different characteristics because, in real environments, noise or interfering sources may not be homogeneously distributed. In order to take into account this aspect, we prefer to consider independently the forward and the reverse channel. Once a mathematical model for the connection in one direction is defined, the reverse connection can be modeled in the same way, possibly with different model parameters. In the following the forward direction only is considered since, in the experiments, data flows mainly from master to slave, while the reverse link is used only for carrying the low control traffic.

The devices used for the measurements do not allow to directly measure the strength of the received signal in terms of signal to noise ratio (SNR). Consequently, the channel condition is expressed in terms of *Packet Dropping Probability (PDP)*, defined as the probability that a packet is discarded due to AC or HEAD detection failure. This metric is independent of the packet format used because the structure of the AC and HEAD fields is exactly the same for all the ACL packets provided by Bluetooth. Therefore *PDP* reflects the real channel condition only and it turns out to be suitable as signal strength indicator.

Besides the *PDP*, we consider the *Payload Error Probability (P<sub>CRC</sub>)*, i.e. the probability of unrecoverable errors in the payload field, given that the AC and HEAD fields are accepted. Notice that, as opposed to *PDP*, *P<sub>CRC</sub>* is determined by both radio channel conditions and packet format used. In fact, channel conditions being equal, packets with different payload length and error protections can experiment different *P<sub>CRC</sub>*.

*PDP* and *P<sub>CRC</sub>* determine the *Packet Error Probability (PEP)*, which is defined as the generic probability of a bad packet reception (due to whatever cause) and is given by

$$PEP = PDP + (1 - PDP)P_{CRC} . \quad (IV.1)$$

A measure of the average capacity perceived by the upper layer applications is given by the *Goodput (G<sub>p</sub>)*. This parameter is defined as the average number of data bits successfully transmitted in forward direction per unit time of time. *G<sub>p</sub>* is realized by the base-band packets that are received with correct AC, HEAD and CRC fields. It is upper

bounded by the *link capacity* ( $C_p$ ), which is defined as the total amount of data bits that can be sent per unit of time.  $C_p$  depends only on the packet format used in each direction and is, therefore, independent of channel conditions.  $G_p$  and  $C_p$  are related by the following expression:

$$G_p = C_p (1 - PEP). \quad (\text{IV.2})$$

### IV.2.C. Statistic confidence & data analysis methodology

The performance metrics introduced in the previous subsection coincide, basically, with the probability of the various error events that may affect the packet reception. Let  $E_k$  denote a generic packet error event:

$$E_k = \{\text{the } k\text{-th packet sent is affected by errors}\}.$$

Then, the packet error process at the receiving side can be represented as a discrete-time binary random process  $x(k)$  defined as

$$x(k) = \begin{cases} 1 & E_k \text{ occurs;} \\ 0 & \text{otherwise.} \end{cases} \quad (\text{IV.3})$$

Assuming that the process  $x(k)$  is wide-sense stationary, the probability  $P$  of the event  $E_k$  is given by the statistic average of the process  $x(k)$ , i.e.:

$$P = E[x(k)], \quad (\text{IV.4})$$

where  $E[\cdot]$  denotes the statistical expectation function. Furthermore, assuming the ergodicity of the process  $x(k)$ , an estimation  $\tilde{P}$  of  $P$  may be obtained by dividing the number of erroneous packets that have been received over the total number of packet that have been sent

$$\tilde{P} = \frac{1}{K} \sum_{k=0}^{K-1} x(k). \quad (\text{IV.5})$$

It is well known that as  $K$  tends to infinity,  $\tilde{P}$  tends to a Gaussian-distributed random variable with

$$\begin{cases} \text{mean} & m = P; \\ \text{variance} & \sigma^2 = \frac{P(1-P)}{K}. \end{cases} \quad (\text{IV.6})$$

The accuracy of the probability estimation, therefore, depends on the number  $K$  of observed samples: the greater is  $K$ , the better is the estimation. The reliability of the estimation can be expressed in terms of *statistic confidence* ( $P_{conf}$ ), defined as the probability that  $\tilde{P}$  belongs to a given confidence interval  $CI = [P_L, P_U]$ . Denoting with  $\varphi(\cdot)$  and  $\Phi(\cdot)$  the normalized Gaussian pdf (probability density function) and cdf (cumulative distribution function) [62], respectively, the statistic confidence  $P_{conf}$  can be expressed as

$$\begin{aligned} P_{conf} &= \Pr[P_L \leq \tilde{P} \leq P_U] = \Pr[\tilde{P} \in CI] \\ &= \frac{1}{\sigma} \int_{P_L}^{P_U} \varphi\left(\frac{a-m}{\sigma}\right) da = \Phi\left(\frac{P_U-m}{\sigma}\right) - \Phi\left(\frac{P_L-m}{\sigma}\right) \end{aligned} \quad (\text{IV.7})$$

In the rest of the analysis we consider the following family of confidence intervals:

$$CI = [(1-a)P, (1+a)P], \quad (\text{IV.8})$$

with  $0 \leq a \leq 1$ . Substituting (IV.6) and (IV.8) in (IV.7) we obtain

$$\begin{aligned} P_{conf} &= \Phi\left(\frac{P(1+a)-P}{\sqrt{P(1-P)/K}}\right) - \Phi\left(\frac{P(1-a)-P}{\sqrt{P(1-P)/K}}\right) \\ &= \Phi\left(a\sqrt{\frac{PK}{(1-P)}}\right) - \Phi\left(-a\sqrt{\frac{PK}{(1-P)}}\right) \\ &= 2\Phi\left(a\sqrt{\frac{PK}{(1-P)}}\right) - 1, \end{aligned} \quad (\text{IV.9})$$

where, in the third step, we used the relation  $\Phi(-a) = 1 - \Phi(a)$ .

Given the confidence interval  $CI$  and the probability  $P$ , we are interested on determining the minimum number of observations  $K$  for which the statistic confidence of  $\tilde{P}$  results greater than, or equal to, a given threshold  $\bar{P}$ . In other words, we want to determine the value  $K(\bar{P})$  such that, for each  $K \geq K(\bar{P})$ ,  $P_{conf} = \Pr[\tilde{P} \in CI] \geq \bar{P}$ . Using (IV.9) in the previous inequality, we obtain

$$\Phi\left(a\sqrt{\frac{PK}{1-P}}\right) \geq \frac{1+\bar{P}}{2}. \quad (\text{IV.10})$$

Since  $\Phi(\cdot)$  is a monotonically increasing function, then it can be (numerically) reversed. Denoting with  $\Phi^{-1}(\cdot)$  the inverse function of  $\Phi(\cdot)$ , it is easy to show that the inequality (IV.10) is satisfied by each integer  $K$  such that

$$K \geq K(\bar{P}), \quad (\text{IV.11})$$

with

$$K(\bar{P}) = \left(\frac{u(\bar{P})}{a}\right)^2 \frac{1-P}{P}; \quad (\text{IV.12})$$

$$u(\bar{P}) = \Phi^{-1}\left(\frac{1+\bar{P}}{2}\right). \quad (\text{IV.13})$$

Therefore, once fixed the desired statistic confidence of the estimation and the confidence factor  $a$ , (IV.12) gives the minimum number of observations that are needed to satisfy the requirements.

The data collected are analyzed on the basis of the previous argumentation. As already introduced, channel conditions are expressed in terms of packet dropping probability ( $PDP$ ) and the other metrics are plotted against this indicator. Hence, the  $PDP$  estimation is required to be fairly accurate. Tab. IV.1 summarizes the parameters considered during the analysis.

TAB. IV.1 STATISTIC CONFIDENCE PARAMETERS CONSIDERED

Statistic confidence	$P_{conf} \geq \bar{P} = 0.95$
Confidence Interval	$CI = [(1-a), (1+a)] \cdot PDP;$ $a = 0.2$

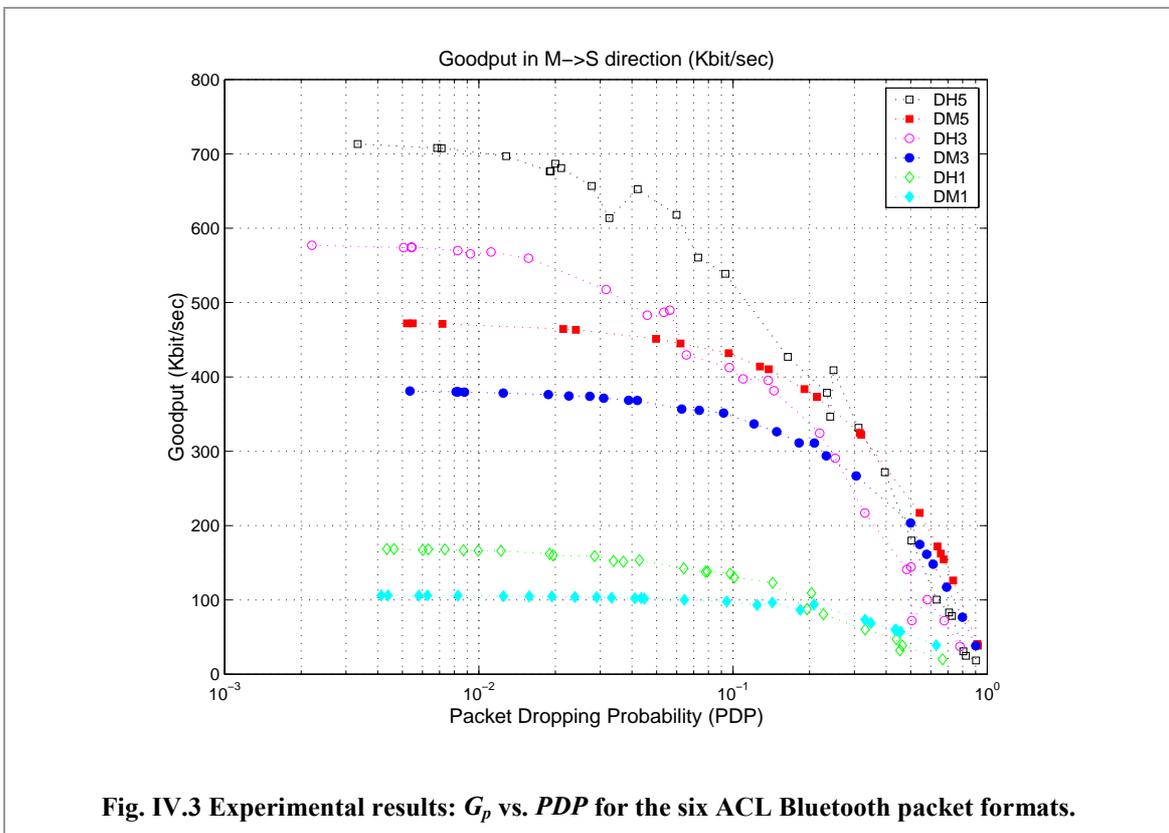
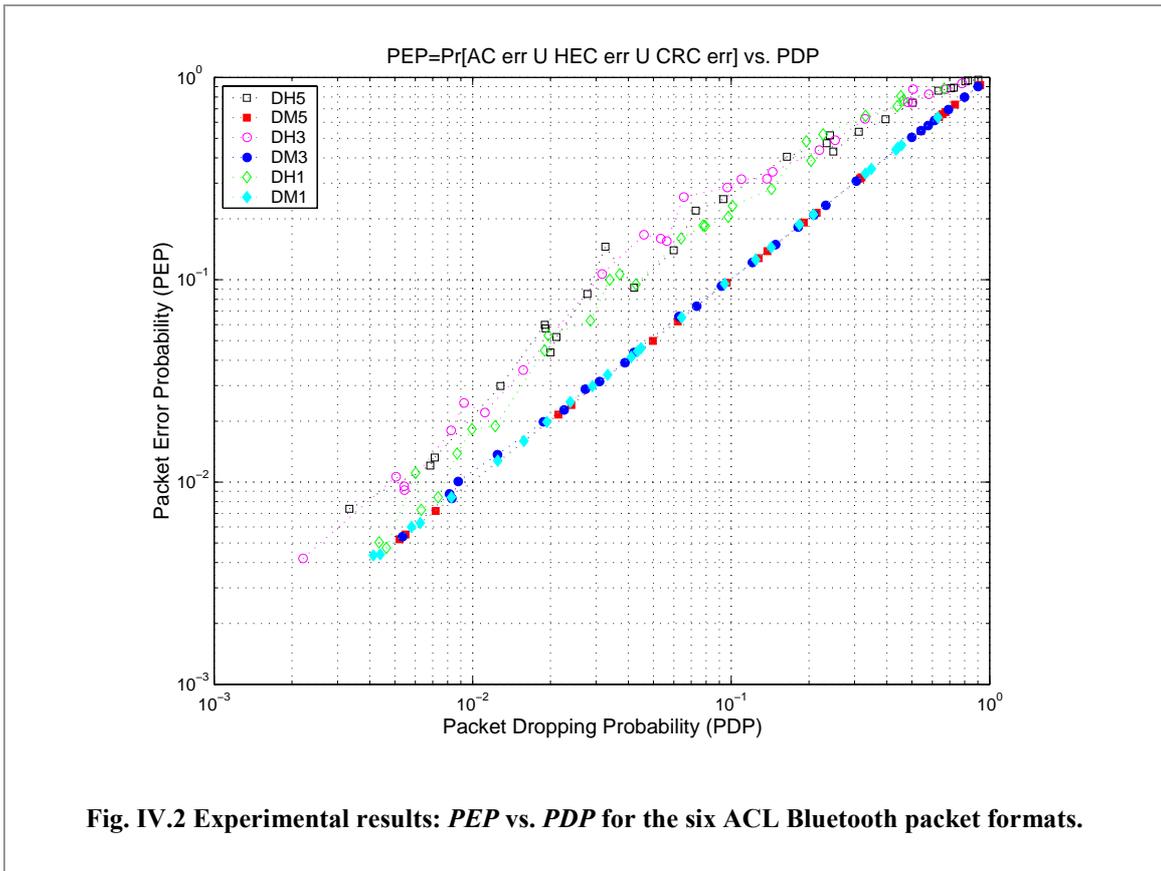
#### IV.2.D. Experimental results

In this section we present and analyze the results obtained by experimental measurements.

Protecting payload with FEC produces two opposite effects: on one hand, the FEC theoretically improves the  $P_{CRC}$  and lowers the  $PEP$ ; on the other hand, the code overhead reduces the payload capacity  $C_p$ . Thus a trade-off between goodput realized by DMn and DHn packets may be expected.

Fig. IV.2 shows the  $PEP$  curves for the six ACL packet formats supplied by Bluetooth, versus the  $PDP$  value that has been measured. We can notice that  $P_{CRC}$  realized by protected formats (DMn) is rather negligible in comparison with  $PDP$  and so  $PEP$  roughly coincides with  $PDP$ . On the contrary, unprotected packet formats (DHn) present significant  $P_{CRC}$  values that result in  $PEP$  values higher than  $PDP$  ones. Consequently, the FEC code appears to be able to correct almost all the errors in the payload field.

Notwithstanding, we have found that unprotected formats achieve better performance than protected ones in almost all the situations that we have considered. Observing the  $G_p$  curves plotted in Fig. IV.3, we can note that DHn packets realize a higher  $G_p$  than DMn packets in almost all the cases, unless under particularly hostile channel conditions. Only when  $PDP$  exceeds roughly  $10^{-1}$ , the FEC protection may be useful in order to maximize the average Goodput of the link.



Such a result may be partially explained by the following observations. The payload field of a packet is considered only after the access code (AC) and the packet header (HEAD) fields have been correctly received. Since the radio channel is slowly time-variant, if the received signal is sufficiently strong to permit the correct reception of AC and HEAD fields, then very likely it will be strong enough to guarantee a low error probability in the payload field as well. Consequently the FEC benefit appears to be unable to compensate the loss of efficiency due to the protection overhead.

Another interesting evidence emerges from the comparison of long and short packet formats. As already mentioned in Chapter III, the carrier frequency remains unchanged for the whole duration of a packet transmission. Therefore, multi-slot packets have higher payload capacity since they reduce the efficiency loss due to the PLL settling time. On the other hand, if an error occurs anywhere in the payload field, the entire packet must be retransmitted. Thus, channel condition being equal, long packets should have higher packet error probability than short ones.

However, measurements show that unprotected packets with different payload length realize merely the same *PEP*, at least for *PDP* values greater than  $10^{-2}$ . These results suggest a model in which an *accepted packet* may experience two different situations: one characterized by a very low bit error rate (BER), and the other with a BER such that DHn-payload contains errors with a probability close to 1, regardless of the packet length. At the same time, since DM packets are almost never affected by payload errors, the BER in the worst case should be small enough to assure a good payload recovery when FEC is used.

On the basis of such observations, we develop a mathematical model for the Bluetooth radio link.

### **IV.3. Modeling Bluetooth radio link**

On the basis of the measurement results, we define a simple mathematical model for the Bluetooth radio link. Such a model should satisfy some requirements. First, it should reproduce the error patterns at the receiving side independently of the radio packet format used. In other words, the parameters that define the model should be derived from the average radio channel conditions only, not from the packet type considered.

Second, the model should be as simple as possible, in order to reduce the computational resources it requires and the complexity of its implementation. Third, it should be able to reproduce, with good accuracy, the performance that each one of the six packet formats achieves at varying of the average channel conditions.

Markovian models appear particularly suitable for our purposes, on account of their simplicity and capability to reproduce, with good accuracy, the error process at the receiving side of a radio link. In the following we discuss how the parameters that define a general Markovian radio channel model, as defined in Section II.2, have been chosen to suit the specific case of the Bluetooth radio channel.

### IV.3.A. Number of states

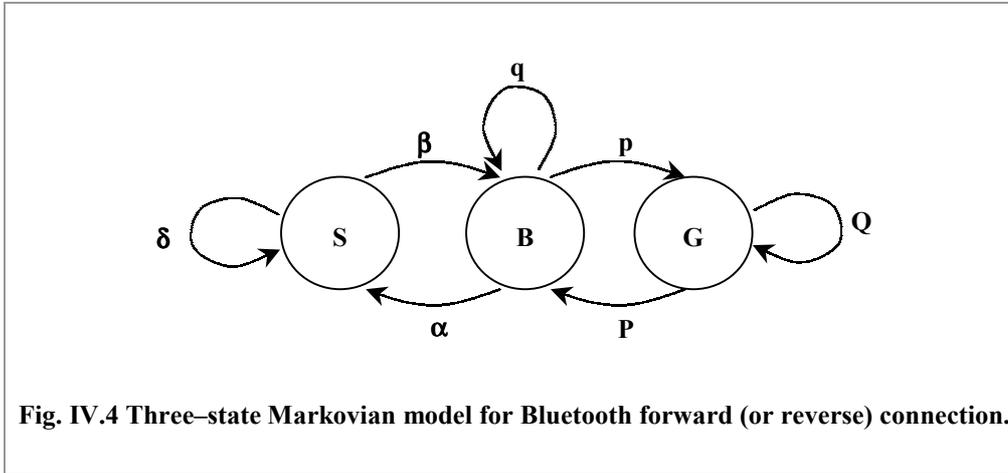
In general, as the number of states increases the capability of the model to reproduce a given error pattern improves, but at the same time, the simulation cost increases. In order to define a general and simple model for Bluetooth radio link in real environment, we have considered the minimum number of states that realizes a good matching between experiment and simulation results, for almost all the interesting cases. Thus, we have considered the following set of states:  $\Sigma = \{S, B, G\}$ , where

- $S$  is the *Synchronization Failure* state, representing the event of a packet received with unrecoverable errors in the AC or HEAD fields;
- $B$  is the *Bad* state, characterized by a not negligible bit error probability;
- $G$  is the *Good* state, representing an error-free reception.

### IV.3.B. State transition step

The state transition step  $T_s$  is given a value equal to the Bluetooth bit duration  $T_b=1\mu\text{s}$ . With this choice, the model parameters are independent of the packet format used. In fact, by choosing a longer transition time, each step will extend over many bits and a different error probability in the  $B$  state should be considered for coded and uncoded formats. Furthermore, at the carrier wavelength used by Bluetooth (around 12.5 cm), the maximum Doppler frequency  $f_m$  is about 16 Hz, or equivalently the minimum coherence time of the channel is 62.5 ms. Since  $T_b$  is much shorter than the coherence time, we may assume that the fading process has very little change at each step. Consequently,

setting the transition step to  $T_b$ , we can fairly safely assume that consecutive transitions occur only among neighboring states, since it is very unlikely that the signal level varies drastically in a step time.



### IV.3.C. State transition matrix

Fig. IV.4 shows the Markovian channel model considered. In the following, the notation adopted by Gilbert in [32] will be used, in order to facilitate the use of some results derived by Gilbert.

The model evolves through continuous transitions among the three states, governed by the state transition probability matrix  $\mathbf{T}$ , which depends on the signal to noise ratio (SNR) thresholds chosen and statistical properties of the fading process. Following the theory presented in Section II.2.A, we denote with  $\Gamma(t)$  the signal to noise ratio experimented by the receiver at the generic time  $t$ , and with  $\Lambda_1$  and  $\Lambda_2$  the thresholds that split the SNR range in three regions. Thus, at the generic instant  $t$  the system state is:

- S if  $\Gamma(t) \in R_S = [0, \Lambda_1)$ ,
- B if  $\Gamma(t) \in R_B = [\Lambda_1, \Lambda_2)$ , and
- G if  $\Gamma(t) \in R_G = [\Lambda_2, \infty)$ .

The state transition probabilities can be computed by using the approximation (II.42) and (II.43) while (II.40) can be used to obtain the steady state probability vector:

$$\mathbf{p} = [P_S, P_B, P_G]. \quad (\text{IV.14})$$

This vector can serve as the set of initial state probabilities each time the carrier frequency changes. Indeed, assuming the wide sense stationary uncorrelated scattering (WSSUS) fading model, we can consider a “new” Markov chain (MC) at each frequency hop. The evolution of this MC starts from a given state with a probability equal to the asymptotic probability of such a state.

It may be worthwhile to notice that Bluetooth provides a limited number of frequency carriers. Thus, after a finite time interval, the same carrier has to be reused. Therefore, an interesting issue is regarding the residual correlation between the states of the MC in two consecutive packet transmissions over the same carrier, since it may be possible that the fading process keeps memory of its history over a reusing time  $T_R$ . In the following we assume an independent MC for each packet transmission, referring to Appendix IV-A for the rationale behind this assumption.

#### IV.3.D. Error probability vector

In the considered model, each state is associated with a binary symmetric channel, with a specific crossover probability given by (II.44). The modulation technique used by Bluetooth is a Gaussian Frequency Shift Keying (GFSK) with a factor  $B_f T = 0.5$  and a modulation index  $2T_b f_c$  between 0.28 and 0.35 ([70],[81]). Unfortunately, we do not have yet a closed expression for the bit error probability in function of SNR, for a transmission scheme with these parameters. As a consequence, the choice of the 1x3 error probability vector  $\mathbf{e} = [e_S, e_B, e_G]$  has to be made on an empiric base. Since the state  $S$  has been assigned to a situation of packet dropping due to AC or HEAD detection failure, we set the crossover probability in this state to the highest value, i.e.  $e_S = 1$ . On the contrary, the state  $G$  is assumed to be error-free, so that  $e_G = 0$ . Finally, the crossover probability in  $B$  state has been chosen on the basis of the observation about the  $P_{CRC}$  values realized by different packet formats (Section IV.2.D). In accordance to that discussion,  $e_B$  should be chosen small enough to guarantee the perfect reception of protected packets, but sufficiently high to assure a payload error probability close to 1 for unprotected formats. Considering the error correcting

capability of the FEC code defined for the DMn format and the payload length of different packet formats (see Section III.6), a good compromise can be reached by choosing  $e_b = 2.5e-3$ .

### IV.3.E. Parameters matching

Once the mathematical model has been defined, we need a method for deriving its parameters. Gilbert, Elliot and Fritchman proposed techniques based on the error-free run distribution ([32], [24],[28]). Unfortunately, these techniques cannot be used in our context, since we do not have the possibility of extracting this distribution from our measurements. Instead, we derive the expressions of the error probabilities that we have defined in Section IV.2.B directly from the mathematical model considered. Successively, we choose the threshold values  $\Lambda_i$  in order to maximize the matching between experimental and the analytical results.

In the following we derive the analytical expressions for error probabilities and present the results obtained by simulation.

### IV.3.F. Packet Dropping Probabilities: *PDP*

At the beginning of each packet transmission the channel model can be in one of the three states, according to the steady state probability of each one. We assume that a data packet is discarded when the SNR at the receiving side  $\Gamma(t)$  belongs to the range  $R_S$ , as defined in 0, and consequently it does not exceed the minimum threshold  $\Lambda_1$ . In other words, a packet is dropped if the initial channel state is S. Then, considering Rayleigh fading, we can express PDP as

$$PDP = P_s = P_r[\Gamma \leq \Lambda_1] = 1 - e^{-\frac{\Lambda_1}{\Gamma}} \quad (\text{IV.15})$$

from which we can derive

$$\frac{\Lambda_1}{\Gamma} = -\ln(1 - PDP). \quad (\text{IV.16})$$

Notice that PDP is the index of the average channel condition and then it is assumed to be an input parameter of the model. Therefore, (IV.16) gives one of the two ratios we need to derive all the model parameters.

### IV.3.G. Payload Error Probability: $P_{CRC}$

#### *Unprotected packet formats: DH*

To derive the expression of  $P_{CRC}$  for unprotected packet formats, let us introduce some notation. Let  $u_G(N)$  and  $u_B(N)$  be the probabilities that no error appears in a sequence of  $N$  bits starting from  $G$  and  $B$  state, respectively. The probability that an accepted DHn packet does not have any error in the payload field is then given by:

$$1 - P_{CRC} = \frac{P_G \cdot u_G(N) + P_B \cdot u_B(N)}{1 - P_S} = \overline{P}_G \cdot u_G(N) + \overline{P}_B \cdot u_B(N), \quad (\text{IV.17})$$

where  $N$  is the payload length (in bits), while  $\overline{P}_G$  and  $\overline{P}_B$  are the steady state probabilities of states  $G$  and  $B$ , respectively, given that the channel state is not  $S$ .  $\overline{P}_G$  and  $\overline{P}_B$  are given by

$$\overline{P}_G = \frac{P_G}{1 - P_S}, \quad (\text{IV.18})$$

$$\overline{P}_B = \frac{P_B}{1 - P_S}. \quad (\text{IV.19})$$

In [32], Gilbert gives a closed form for  $u_B(N)$  in the case of his two-state model. Following a formally analogous procedure, we can obtain a similar expression for  $u_B(N)$  even in the three-state model considered in this chapter, except that in this case, the probabilities  $p$  and  $q$  are not complementary. Indeed, in order to have an error-free sequence, transitions can occur only between state  $G$  and  $B$ , as in the Gilbert model, since the error probability in state  $S$  is set to 1. Furthermore, with  $e_G = 0$ , the expression

for  $u_G(N)$  can be easily derived in the same way as  $u_B(N)$ . Referring to [32] for the details, we consider directly the final expressions for these probabilities given by

$$u_G(N) = \frac{(J + h_B(pP - Qq))J^N - (L + h_B(pP - Qq))L^N}{J - L}; \quad (\text{IV.20})$$

$$u_B(N) = \frac{(J + pP - qQ)J^N - (L + pP - Qq)L^N}{J - L}; \quad (\text{IV.21})$$

where  $h_B = 1 - e_B$  is the probability of no error in state  $B$ , while  $J$  is

$$2J = Q + h_B q + \sqrt{(Q + h_B q)^2 + 4h_B(pP - qQ)} \quad (\text{IV.22})$$

and  $L$  is the same expression with negative square root.

### ***Protected packet formats: DM***

Unfortunately, so far we have not found a closed expression for the payload error probabilities for protected formats. Instead, we derived a recursive method for their computation. It is based on the H(15,10) shortened Hamming code used by Bluetooth, able to correct up to one erroneous bit into a 15-bits codeword.

Let us introduce, for every  $X, Y \in \Sigma = \{S, B, G\}$ , the following notation:

- $S_X = \{\text{the initial state of the MC is } X\}$ ;
- $L_X = \{\text{the final state of the MC is } X\}$ .
- $v_X(n) = \Pr[n \text{ codewords are well recognized } | S_X]$
- $r_{XY}(n) = \Pr[\text{a codeword is well recognized, } L_Y | S_X]$
- $w_{XY}(i, j) = \Pr[i \text{ errors in a sequence of } j \text{ bits, } L_Y | S_X]$

It is easy to show that, for  $n \geq 1$  and  $\forall X \in \Sigma$ , we have

$$v_X(n) = \sum_{K \in \Sigma} r_{XK} v_K(n-1) \quad (\text{IV.23})$$

while, for  $n=0$  we assume  $v_X(0)=1$ . Furthermore, due to the one-bit correction capability of the Hamming code used by Bluetooth protected packet formats,  $r_{XY}$  can be expressed as

$$r_{XY} = w_{XY}(0,15) + w_{XY}(1,15); \quad \forall X, Y \in \Sigma \quad (\text{IV.24})$$

Also these probabilities can be computed recursively. Taking advantage of the notations introduced in Section II.2.A for the state transition probabilities ( $t_{XY}$ ) and the per-state crossover probabilities ( $e_X$ ), we have

$$\begin{aligned} w_{XY}(0, m) &= \sum_{K \in \Sigma} t_{XK} (1 - e_K) w_{KY}(0, m-1), \quad m \geq 1 \\ w_{XY}(1, m) &= \sum_{K \in \Sigma} t_{XK} [(1 - e_K) w_{KY}(1, m-1) + e_K w_{KY}(0, m-1)], \quad m \geq 1 \\ w_{XY}(0, 0) &= \delta_{X,Y}; \\ w_{XY}(1, 0) &= 0; \end{aligned} \quad (\text{IV.25})$$

where  $\delta_{XY}$  is 1 if  $X=Y$  and 0 otherwise. Therefore, the probability of realizing a good reception of a DM packet with a payload of  $N$  bits is approximately given by

$$1 - P_{CRC} = P_G \cdot v_G(M) + P_B \cdot v_B(M) + P_S \cdot v_S(M), \quad (\text{IV.26})$$

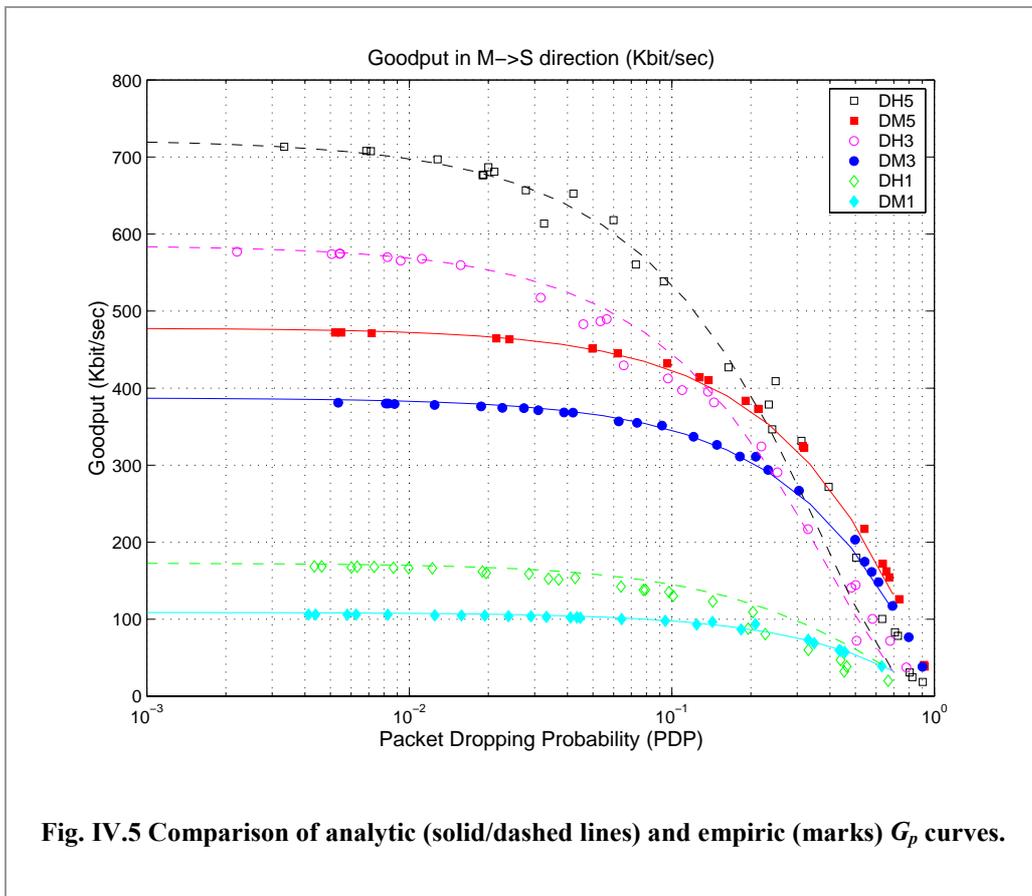
where  $M$  is the number of codewords in the payload field, so that  $M \approx N/15$ .

### IV.3.H. Validation of the mathematical results

By using (IV.15), (IV.17) and (IV.26) we can estimate  $P_{CRC}$  versus  $PDP$  for all the six Bluetooth packet formats. Therefore, we can choose  $\Lambda_1$  and  $\Lambda_2$  in order to maximize the matching between analytic and empiric results. Fig. IV.5 shows the analytic curves compared with the experimental ones, for  $\Lambda_1=1$  and  $\Lambda_2=2.5$ . Generally,  $G_p$  obtained from the model matches fairly well with the measurement results, in particular for multi-slots packet formats. We can see that the experimental results for DM5 and DM3 formats are slightly underestimated. This is due to the non-zero possibility that the MC visits the state  $S$  during the ‘‘reception’’ of the payload field, experiencing a drastic

increase in the error probability. This behavior does not correspond to a real situation in which the error probability changes smoothly during the reception of a packet. Nevertheless, the simplification in the analysis that derives by the assumption of  $e_s = 1$  compensates for the small mismatch of the results.

Finally, we note that the gap between estimated and empirical DH5 performance grows for PDP greater than 0.2. For such PDP values, however, the link is practically useless and hence this mismatch does not compromise the “goodness” of the mathematic model that has been proposed.



## IV.4. Conclusions

Concluding, in this chapter we have presented the results of a set of measurements of an FTP connection over a single-hop Bluetooth radio link. The measurement-based approach considered gave us the opportunity to observe the system behavior in real-world situations and thus to analyze the performance obtained by using different Bluetooth packet formats. Measurements have revealed that protected packet formats

suffer from the inefficiency of FEC overhead that is not compensated by an adequate improvement of packet retransmission probability. Furthermore, in almost all the situations that have been considered, long packets appear more performing than short ones in terms of goodput.

Furthermore, from the collected data we have derived a simple three-state Markovian model for the wireless link that captures the aggregate of real-world effects like noise, interference and fading. In spite of its simplicity, the model matches fairly well with the measures, and proves to be suitable for simulating the link behavior for all the six packet formats supplied by Bluetooth.

## APPENDIX IV-A

### MARKOV CHAINS MEMORY OVER A REUTILIZATION TIME.

Let  $T_R$  be the average carrier reutilization time, i.e. the average time elapsed between two consecutive transmissions over the same carrier frequency, in the same direction. If  $T_R$  is much longer than the average fade/non-fade cycle duration we can assume that, in this time, the fading process loses memory. In other words, when the transmitter utilizes a specific frequency for the second time, the state of the channel can be assumed independent of the state at the previous utilization.

A measure of correlation between states separated by  $N_R = T_R/T_s$  steps, can be provided by the *Average Mutual Information* between the state of the MC at a generic step  $n$  and  $N_R$  steps later ([76]). Let  $X_i$  be the state of the MC at the  $i$ -th step. Then the information provided by the occurrence of the event  $X_i = K$  about the event  $X_{i+n} = Y$ , with  $K, Y \in \Sigma$ , is given by

$$I(Y; K) = \log \frac{\Pr(X_{i+n} = Y | X_i = K)}{\Pr(X_{i+n} = Y)} \quad (\text{IV-A.1})$$

The average value of the mutual information can be obtained by simply weighting  $I(Y; K)$  over all the possible joint events. Thus the average mutual information between  $X_i$  and  $X_{i+n}$  is given by

$$\begin{aligned} I(X_{i+n}; X_i) &= \sum_{K \in \Sigma} \sum_{Y \in \Sigma} \Pr(X_{i+n} = Y, X_i = K) I(Y; K) = \\ &= \sum_{K \in \Sigma} \sum_{Y \in \Sigma} \Pr(X_{i+n} = Y | X_i = K) \Pr(X_i = K) \log \frac{\Pr(X_{i+n} = Y | X_i = K)}{\Pr(X_{i+n} = Y)} \end{aligned} \quad (\text{IV-A.2})$$

Remembering the properties of Markov chains, the probabilities that appear in (IV-A.2) can be expressed in terms of state transition probabilities. Indeed, we have:

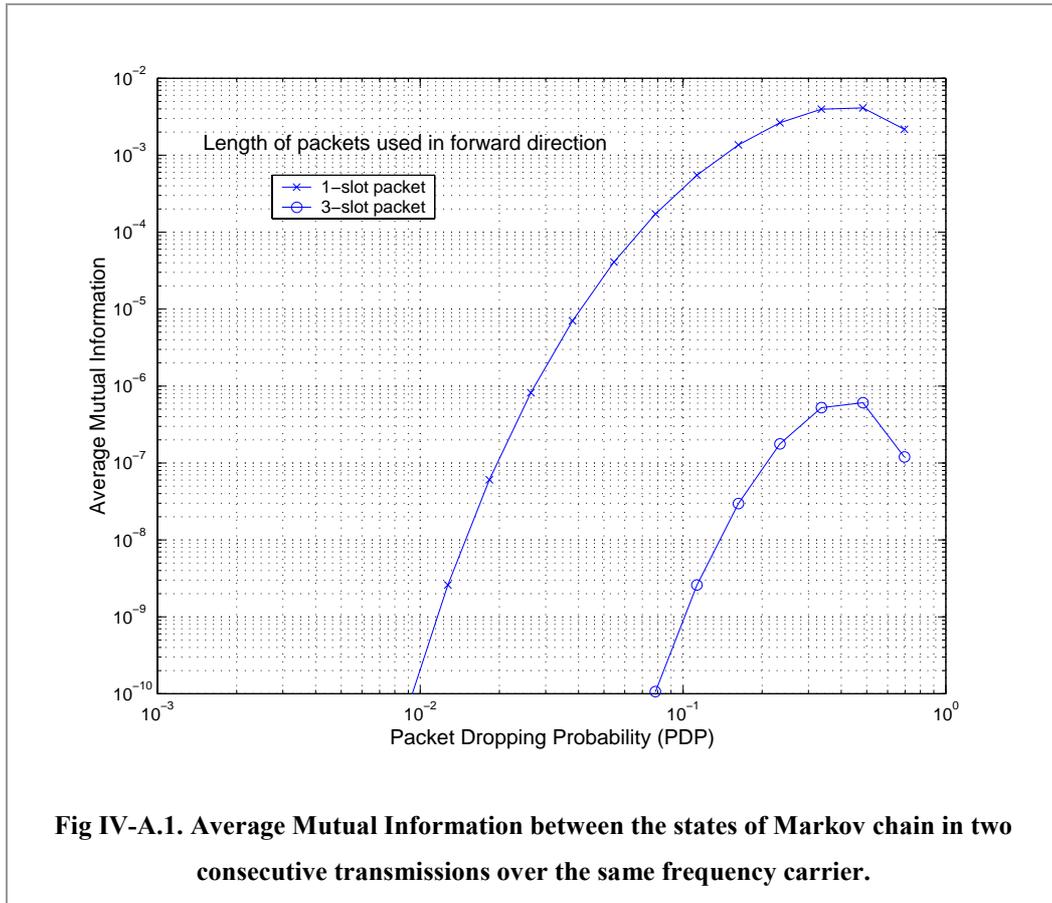
$$\Pr(X_{i+n} = Y | X_i = K) = [\mathbf{T}^n]_{(K,Y)} \quad (\text{IV-A.3})$$

$$\Pr(X_{i+n} = Y) = P_Y \quad (\text{IV-A.4})$$

where  $\mathbf{T}^n$  denotes the  $n$ -th power of matrix  $\mathbf{T}$ . Then, equation (IV-A.2) becomes

$$I(X_{i+n}; X_i) = \sum_{K \in \Sigma} \sum_{Y \in \Sigma} [\mathbf{T}^n]_{(K,Y)} P_K \log \frac{[\mathbf{T}]_{(K,Y)}}{P_Y} \quad (\text{IV-A.5})$$

Note that, as the correlation between symbols  $X_i$  and  $X_{i+n}$  decreases, the average mutual information  $I(X_{i+n}; X_i)$  tends to zero.



**Fig IV-A.1. Average Mutual Information between the states of Markov chain in two consecutive transmissions over the same frequency carrier.**

Fig IV-A.1 shows the average mutual information between two states of the Markov chain in two consecutive transmissions over the same carrier, in the case of single-slot and three-slot packet lengths. We note that the shortest packets, which realize the smallest reutilization time, show average mutual information that remains always below  $1e-2$ . Therefore, we can assume that at each hop, the system returns to a new Markov chain with a state probability vector given by (IV.14).

---

## V. POLLING MECHANISMS FOR BLUETOOTH

---

*The problem of finding an efficient polling algorithm for piconet operating is similar to the problem of centrally-controlled polling schemes. However these algorithms may result inefficient in the particular case of Bluetooth, due to the peculiarities of this technology that introduce an additional degree of difficulties. Moreover, scatternets add new constraints to the scheduling issue. In particular, the management of gateway units, i.e. of units that forward traffic among different piconets, is not trivial. All these constraints will significantly impact the performance of data traffic over Bluetooth. In this chapter classical scheduling schemes are outlined and discussed and a new scheduling scheme for Bluetooth piconets and scatternets is proposed and analyzed.*

### V.1. State of art: related polling schemes

As introduced in Chapter III, two or more Bluetooth devices in a range of approximately ten meters can set up an ad hoc connection and establish a piconet. In each piconet a unit assumes the role of master, controlling the other units that act as slaves. Transmission occurs only between master and slaves, on the basis of a TDD scheme. In each master-to-slave slot, the master can send a data or a POLL packet to a slave, enabling (and soliciting) the slave to reply on the following slave-to-master slot (Section III.3)

The medium access control mechanism adopted by Bluetooth to manage transmissions within a piconet can be modeled as a queue system in which a server visits a set of queues to serve customers, following a given strategy that, generally, is referred to as *polling scheme*. Then, the polling scheme is the set of rules that determines the order of visit of the different queues, the number of customers of a queue to be served, etc.

Many different polling schemes have been proposed and analyzed in literature for centrally-controlled polling systems. In [53], authors prove that the optimal polling schemes for uniformly loaded systems (same traffic offered to all the queues) are exhaustive. In such schemes, the server keeps serving the customers in a queue until it is empty. Moreover, assuming the server knows all the queue lengths, the optimal

performance is achieved serving first the longest queue. On the other hand, when the status of the queues is not known, the simple cyclic routing scheme yields the best performance. The main drawback of these kind of schemes is that they may lead to an unfair distribution of the server capacity. Indeed, on the basis of the exhaustive paradigm, a queue gets more only because it requires more. In such a way, more hungry sources that capture almost the totality of the server capacity may starve the sources with limited capacity demand.

The fairness issue has been kept into consideration in [27], where authors propose a set of polling algorithms that aim to achieve high resource utilization and fairness among users. The polling algorithms impose that each queue must be visited once per cycle, but the visit order can be dynamically changed. For uniformly loaded systems, the optimal performance is achieved considering a decreasing queue length order at the beginning of each polling cycle. This scheme has shown an optimal behavior even for not-uniformly loaded systems, for which, however, the optimality property has not been proved in the general case.

### ***Polling schemes for Bluetooth piconets***

Even though the polling scheme adopted in the Bluetooth piconets may recall a classical centrally-controlled polling scheme, the peculiarities of the Bluetooth technology introduce an additional degree of difficulties that makes the classical algorithms rather inefficient. The constraints added by Bluetooth are the followings:

- i) the master unit can have only a limited information of the queue at each slave;
- ii) a slave unit is allowed to transmit data only if directly addressed by the master in the previous time slot;
- iii) any time the master uses a no payload packet to poll a slave, or a polled slave has no data to send, a slot gets wasted;
- iv) the scheduling mechanism must be kept as simple as possible in order to satisfy the low cost and low memory occupancy objectives.

In particular, the master station may only know if it has any outstanding packets, but it can not know precisely whether a slave has packets, nor how many packets there are in the queue at each slave. These constraints determine a loss of performance due to many reasons:

- It is not possible to assume a convenient tagging mechanisms at master. Thus, the algorithms based on the knowledge of the queues length cannot be properly applied and can achieve only sub-optimal performance.
- In order to monitoring the queue status of a slave, the master should periodically send a packet to it, soliciting the reply. If the master-slave pair produces only uplink traffic (from slave to master), the master has to use a no payload packet, wasting bandwidth.

Therefore, researchers have been trying to modify the classic scheduling algorithms in order to suit the Bluetooth peculiarities and improve piconets performance. Bluetooth specification ([70]) suggests the very simple pure round robin (PRR) scheduling mechanism for piconet, in order to satisfy the low cost objective. With PRR, all slaves get polled cyclically, whether they have data to transmit or not. Bandwidth not used by a lightly loaded slave is, then, lost and cannot be used by other slaves. Hence, the PRR scheme, which has been extensively studied in the literature, may yield low throughput in the specific case of Bluetooth piconets, in particular in the case of many slaves with bursty traffic.

In order to overcome this drawback, [43] defines the meaning of active and inactive state for each slave. Active slaves are polled in a round robin manner and each inactive slave is polled in a inter-poll interval regularly to check whether it has become active or not. The polling scheme divides bandwidth among slaves in a more efficient way, but it yields good performance only if the traffic demand of each slave is known in advance.

[29] and [46] assume that a master unit knows whether a slave has data packet to send or not. If both master and slave have no data packet, this pair is not scheduled. In other words, the master only considers those slaves where either the master or slave queue is non-empty. Therefore such a scheme prevents wastage, but it assumes an ideal scenario where the master has updated information on the status of slave queues.

[87] predicts for each slave whether data is available or not and keeps track of the fairness also based on traffic demand or estimated traffic demand of the slave. [13] extends the algorithm in [87] by separating the scheduling of the uplink and downlink transmission into two cycles: uplink polling sub-cycle and downlink polling sub-cycle, and adopting a truncated binary exponential back-off algorithm to control the slave polling rate during an uplink sub-cycle. However, fairness is guaranteed independently for the uplink and the downlink directions. Thus, a slave may get double bandwidth than another if the master also sends data packets to the slave.

In [14], authors consider, beside the PRR polling scheme, other two classical polling schemes, namely the ERR (Exhaustive Round Robin) and the EPM (Exhaustive Pseudo-cyclic Master queue length). In the ERR, as well as in the PRR, units in the piconet are visited on the basis of a fixed cyclic order. However, while in the PRR each unit gets a single chance to transmit at each cycle, the ERR is exhaustive in nature, i.e. the master keeps polling the same slave until both the master and slave queues are empty. The EPM is similar to the ERR but the polling order is dynamically decided at the beginning of each cycle, on the basis of the sum of the master and slave queue lengths. Each master-slave pair is visited exactly once per cycle, in decreasing order of total queue length.

Moreover, [14] presents a new polling scheme, called limited and weighted round robin (LWRR), that is based on the Limited Round Robin (LRR) polling scheme. The LRR scheme has been proposed to avoid the capture problem that can arise by using an exhaustive polling scheme. To achieve this purpose, the LRR scheme sets a limit to the number  $t$  of transmissions (tokens) that can be performed by each master-slave pair in each polling cycle. In such a way, the cycle length may still change cycle by cycle, but within an upper bounded. The LWRR scheme extends the LRR scheme by adopting a weighted round robin algorithm with weights dynamically changed according to the observed queue status. The weight of a master-slave pair is decreased any time the pair does not exchange data during a polling phase, while it is set to the maximum as soon as a data exchange occurs. The rate of visit of a master-slave pairs with low weight is reduced in order to increase the bandwidth utilization. A scheduling algorithm similar to LWRR has been also presented in [21].

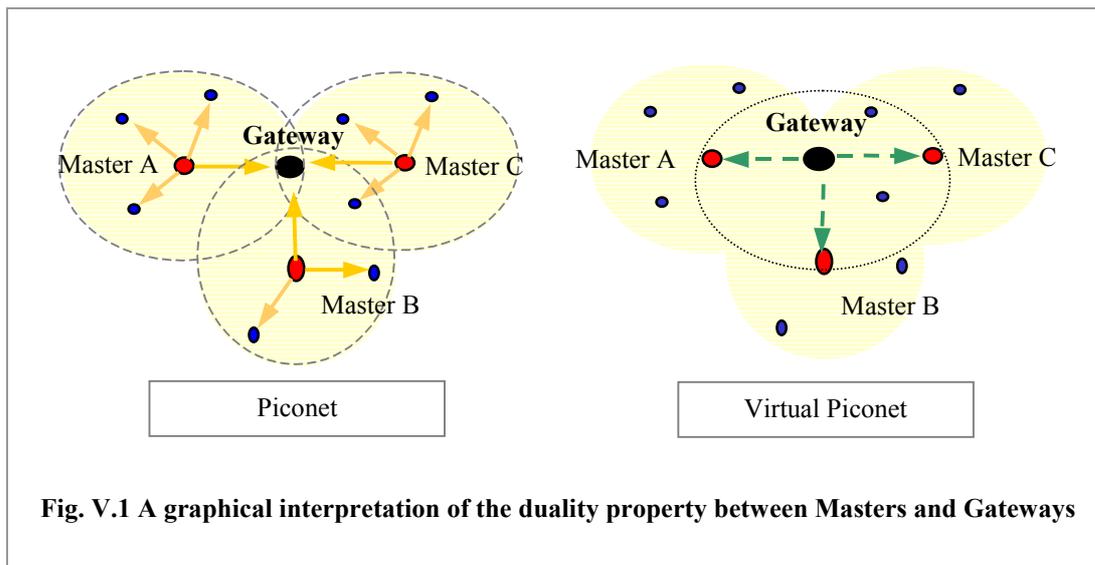
### *Scheduling issue for Bluetooth scatternets*

The scheduling issue becomes more complicated when applied to a scatternet structure. Indeed, the realization of a scatternet requires some units to be present in more than one piconet, in order to forward data among the different piconets they belong to. These units, called gateways, will need to time-division their presence among the piconets. An important issue with the gateways is that their presence in different piconets needs to be scheduled in an efficient manner. Furthermore, since the gateway unit cannot receive information from more than one piconet at a time, there is a need to co-ordinate the presence of masters and gateway devices in each piconet.

## **V.2. An integrated approach**

In the followings of this chapter, we investigate the issue of scheduling in Bluetooth piconets and scatternets. First, we extend the concept of max-min fair resource distribution to a scatternet architecture. In such a way we provide a benchmark for the performance that can be achieved by a fair scheduling algorithm for scatternets. Then, we propose a totally distributed scatternet scheduling algorithm that adapts to non-uniform and changing traffic without requiring any exchange of control information. This algorithm provides an integrated solution for both intra- and inter-piconet scheduling, i.e., for polling of slaves and scheduling of gateways, addressing the issues sketched before. The algorithm aims to improve the utilization of the piconet capacity, as well as the gateway capacity in a scatternet architecture. The algorithm gives a high bandwidth utilization and results in a fair division of a) the piconet bandwidth between the slaves of a piconet and b) the gateway presence among different piconets.

The rest of the chapter is organized as follows. In Section 0 we introduce the duality principle that is at the basis of the proposed algorithm. In Section V.4 a definition of fairness in the context of Bluetooth scatternets, which takes into account intra- and inter-piconet max-min fairness, is given. Section V.5 describes the algorithm and Section V.6 shows simulation results. Section V.7 presents the conclusions.



### V.3. Duality property

As introduced in the previous sections, units belonging to a piconet share the piconet capacity according to the polling algorithm used by the master. In an analogous manner, gateways in a scatternet have to divide their time among the different piconets they belong to, according to the “master–listening” algorithm they use. It can be noted that there is a duality in this architecture. On the one side, each master divides its capacity among the units of its piconet, by using a scheduling algorithm for polling them. On the other side, each gateway shares its capacity among the piconets it belongs to, on the basis of a scheduling algorithm it uses for listening to the masters. The gateway, then, can be viewed as a “virtual master” and its masters can be viewed as “virtual slaves” forming a “virtual piconet”, in which the polling cycle is, actually, the “listening cycle” of the gateway. A graphical interpretation of this duality is given in Fig. V.1.

This duality suggests the use of the same scheduling algorithm for fair sharing of both a) the piconet capacity among slaves and b) the gateway time among piconets. Before describing how the bandwidth of a scatternet may be divided among units in a max–min fair manner, we give the definition of max–min fairness for piconets and scatternets.

## V.4. Fair bandwidth allocation

As introduced in the previous section, all the units in a piconet share the piconet bandwidth on the basis of the scheduling algorithm adopted by the master. This algorithm may distribute the resource among the users following a “max–min fairness” criterion. In such a way each user gets the maximum share of the resource it can have without penalizing the other users.

To formally define the concept of max–min fair share for piconets and scatternets we need some notations.

### Notation (Piconets)

- $x$  : number of master units
- $M_i$  :  $i$ -th master unit,  $i \in \{1, 2, \dots, x\}$
- $\mathbf{M}(i)$  : set of slave units connected to master  $M_i$
- $G(u, M_i)$  : resource rate demand of the slave unit  $u$  for the master unit  $M_i$
- $R(u, M_i)$  : max–min resource share given to the slave unit  $u$  by the master unit  $M_i$

Referring to this notation, we formalize the definition of max–min fairness for a single piconet (see, for instance, [54]).

### Definition V.1 (max-min fairness for single piconet)

Consider the piconet formed by the master unit  $M_i$  and the set of connected slave units  $\mathbf{M}(i)$ . The rate allocation  $R(u, M_i)$  among slave units in  $\mathbf{M}(i)$  is max-min fair if:

1) it is feasible, i.e.

$$i. \quad \forall u \in \mathbf{M}(i), \quad 0 \leq R(u, M_i) \leq G(u, M_i)$$

$$ii. \quad \sum_{u \in \mathbf{M}(i)} R(u, M_i) \leq 1$$

- 2) for each unit  $u$ ,  $R(u, M_i)$  cannot be increased (while maintaining feasibility) without decreasing  $R(v, M_i)$  for some other unit  $v$  for which  $R(v, M_i) \leq R(u, M_i)$ .

An equivalent definition is the following. Without loss of generality, let units be ordered for increasing rate demand, so that  $0 \leq G(u_1, M_i) \leq G(u_2, M_i) \leq \dots \leq G(u_S, M_i)$ , where  $S$  is the total number of slave in the piconet. Then, a rates allocation  $R(u, M_i)$  is max–min fair if and only if it satisfies the following proprieties:

- 1)  $\sum_{u \in \mathbf{M}(i)} R(u, M_i) = \min \left\{ 1, \sum_{u \in \mathbf{M}(i)} G(u, M_i) \right\}$
- 2)  $\exists K \in \{0, 1, \dots, S\} \left| R(u_j, M_i) = \begin{cases} G(u_j, M_i); & j = 1, 2, \dots, K \\ R; & j = K + 1, K + 2, \dots, S \end{cases} \right.$
- 3)  $R \geq G(u_K, M_i)$ , with  $R = \frac{1 - \sum_{j=1}^K G(u_j, M_i)}{S - K}$ .

Note that if  $K=S$  then each unit gets the rate it requires, while whether  $K=0$ , all the units share homogeneously the total capacity. On the basis of this definition, Fig. V.3 provides a simple algorithm to compute the max–min fair rate distribution for a piconet, starting from the vector of rates required by units.

While the application of max–min fair concept to a piconet is straightforward, its extension to a scatternet structure deserves some attention. The duality property, which we have introduced in Section 0, allows to considerably simplify the dissertation. Thus, we extend the notation to virtual piconets as follows.

**Notation** (Virtual Piconets)

- $\tilde{x}$  : number of *virtual master* (gateway) units
- $\tilde{M}_i$  :  $i$ -th virtual master unit,  $i \in \{1, 2, \dots, \tilde{x}\}$
- $\tilde{\mathbf{M}}(i)$  : set of *virtual slave* (master) units connected to the virtual master  $\tilde{M}_i$

Therefore, the concept of max–min fair rate distribution can be extended to a scatternet structure in the following way.

**Definition V.2 (max-min fairness for scatternet)**

Given a scatternet with  $x$  master units and  $\tilde{x}$  gateway units, an allocation of rates among the units is max-min fair if the followings conditions are satisfied:

1) *feasibility in each piconet and virtual piconet*

$$i. \quad \forall i \in \{1, 2, \dots, x\} \text{ and } \forall u \in \mathbf{M}(i), 0 \leq R(u, M_i) \leq G(u, M_i)$$

$$ii. \quad \forall i \in \{1, 2, \dots, x\}, \sum_{u \in \mathbf{M}(i)} R(u, M_i) \leq 1,$$

$$iii. \quad \forall i \in \{1, 2, \dots, \tilde{x}\}, \sum_{u \in \tilde{\mathbf{M}}(i)} R(\tilde{M}_i, u) \leq 1,$$

2) *fairness in each piconet and virtual piconet*

iv.  $\forall i \in \{1, 2, \dots, x\}$  and  $\forall u \in \mathbf{M}(i)$ ,  $R(u, M_i)$  cannot be increased without decreasing  $R(v, M_i)$  for some other unit  $v \in \mathbf{M}(i)$  for which  $R(v, M_i) \leq R(u, M_i)$ .

v.  $\forall i \in \{1, 2, \dots, \tilde{x}\}$  and  $\forall u \in \tilde{\mathbf{M}}(i)$ ,  $R(\tilde{M}_i, u)$  cannot be increased without decreasing  $R(\tilde{M}_i, v)$  for some other unit  $v \in \tilde{\mathbf{M}}(i)$  for which  $R(\tilde{M}_i, v) \leq R(\tilde{M}_i, u)$ .

The distribution of max–min fair rates depends upon the set of rate–demand of the units. It may be worthwhile noting that while the rate–demand of a non–gateway slave depends only on the traffic generated between the master and the slave, the rate–demand of a gateway to a given master depends also on the traffic generated to all the other masters the gateway belongs to. In the following sub–sections, we discuss factors that determine the *actual rate–demand* of a gateway to a given master and lead to the division of the scatternet capacity among the units in a traffic–dependent, bandwidth efficient and fair manner.

We call these factors the *Piconet Presence Fraction* and the *Scatternet Presence Fraction*. In following discussion we assume that the gateway cannot be a master in any piconet.

```

Function maxmin_fair(G,  $\mathbf{M}(i)$ ,  $M_i$ )

% G: rate demand function for the slave units
%   connected to the master  $M_i$ 
%   (for the sake of clarity, in the following
%   we omit to indicate  $M_i$  in the functions)
%  $\mathbf{M}(i)$ : set of slave units in the piconet  $M_i$ 

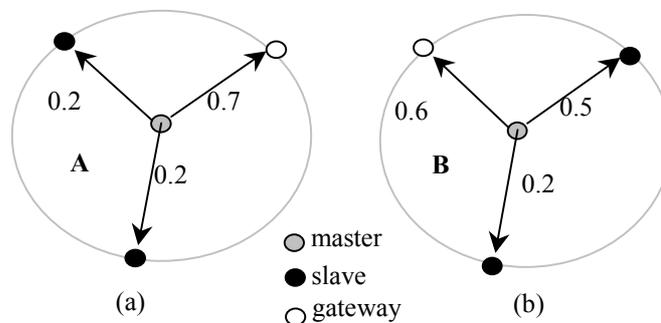
S = size( $\mathbf{M}(i)$ ); % number of slave units
R = 1/S
U = sort( $\mathbf{M}(i)$ ); % sort the unit in ascending order of rate demand

for j=1 to S do, % compute the max-min fair share for each unit
{
    if G( $u_j$ ) < R then
    {
        R( $u_j$ ) = G( $u_j$ )
         $R = \left(1 - \sum_{h=1}^j G(u_h)\right) / (S - j)$ 
    } else
    {
        R( $u_j$ ) = R
    }
}

return R(.)

```

**Fig. V.3** Algorithm for the computation of max–min fair share in a simple case



**Fig. V.2 (a) and (b):** piconets with traffic rates between master and each slave shown

#### V.4.A. Piconet Presence Fraction

Consider a piconet consisting of a gateway slave and some non-gateway slaves in which the master has complete knowledge of the rate demands of all slaves. Using this knowledge, the master polls the slaves in a max-min fair manner giving the gateway a certain fraction of the polling. We call this fraction the “piconet presence fraction” (PPF) of the gateway for this piconet. Note that a gateway has a PPF for each piconet it belongs to.

The importance of the PPF lies in the fact that if the gateway were “perfectly coordinated”<sup>3</sup> with the master, the gateway would need to stay in the piconet for PPF of its total time to get its fair share of polling. It is important to note that the gateway would not gain anything by spending a fraction of time greater than the PPF in a piconet, since the master would not give it a fraction of polling more than the PPF.

Consider the piconets shown in Fig. V.2 (a) and (b), each consisting of one gateway and two slaves, with the traffic rates of each slave as shown. In Fig. V.2 (a) (piconet A), the fair share of each slave is 0.2, while the PPF of the gateway is 0.6. In Fig. V.2 (b) (piconet B), the fair shares of the slaves are 0.2 and 0.4, while the PPF of the gateway is 0.4.

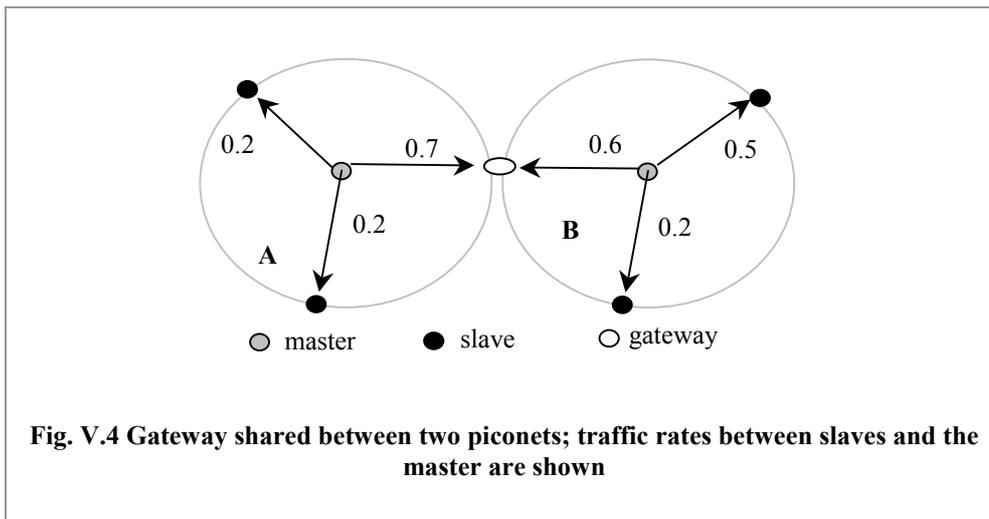
#### V.4.B. Scatternet Presence Fraction

A gateway will, in general, be a slave in multiple piconets and may have different amounts of traffic to exchange with each piconet. Consider an ideal gateway that has complete knowledge of the rate demands of all its masters. The gateway can then divide its presence among the piconets in a max-min fair manner, giving each piconet a certain fraction of its presence. We call this fraction the “scatternet presence fraction” (SPF) of the gateway for the piconet.

The importance of the SPF lies in the fact that it allows the gateway to divide its presence among piconets in a fair manner.

---

<sup>3</sup> Perfect coordination means that the gateway enters a piconet exactly when it is its chance to get polled and leaves the piconet on finishing its polling



Consider the piconets of Fig. V.2 again, but the gateway of each of the piconets now connects them to form a scatternet, as shown in Fig. V.4. The traffic requirements are the same as shown in Fig. V.4. The SPF of the gateway is 0.5 in piconet A and 0.5 in piconet B.

#### V.4.C. Fair Share

We see that there are two forces acting on the gateway: the PPF, which achieves fairness between the gateway and the other slaves of a piconet, and the SPF, which distributes the presence of the gateway between the piconets in a fair manner. The two forces concur to determine the actual rate demand of the gateways in the scatternet.

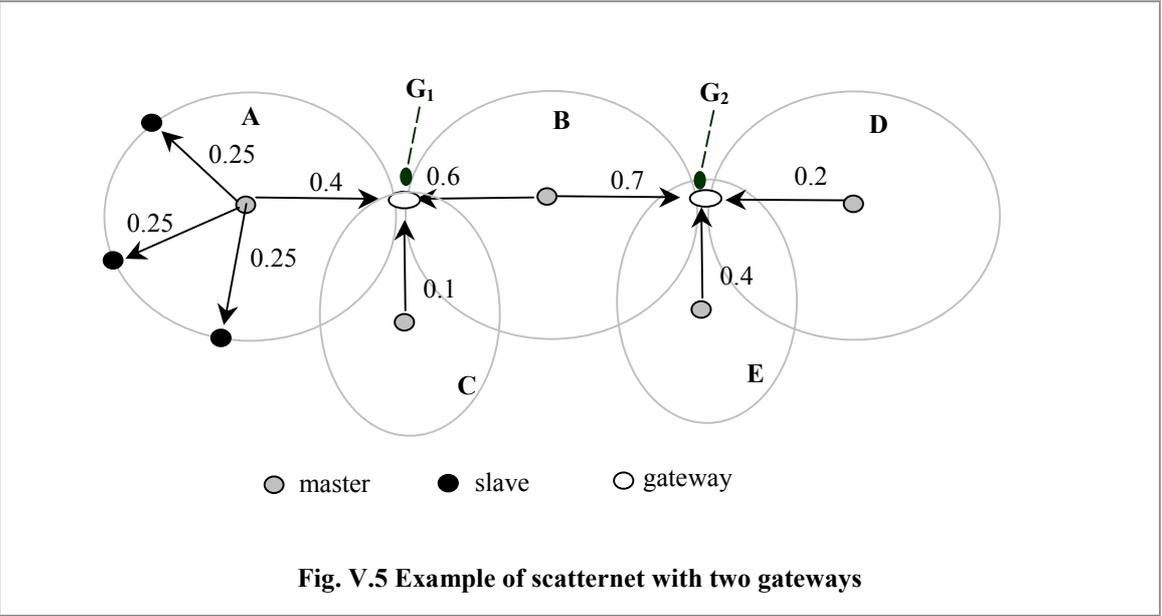
In order to make the concept clear, we first consider the simple case of a scatternet with a single gateway.

Given the rate demand of each unit, the computation of PPF and SPF for the gateway is straightforward. We observe that, once fixed the actual rate demand for every unit, the gateway does not gain anything by spending a fraction of its time greater than the PPF in a piconet since the master does not give it a rate more than the PPF. Thus, if the PPF for a piconet is less than the SPF, the gateway spends a fraction of its time equal to the PPF in the piconet. The extra scatternet presence fraction of this piconet (difference of the SPF and the PPF) may then be redistributed in a fair manner among other piconets for which the SPF is less than the PPF. On the contrary, whether the SPF for a piconet is less than the PPF, the gateway has to spend only a fraction SPF of its time in

the piconet and the extra piconet capacity (the difference between the PPF and the SPF) may be redistributed among the other units in the piconet. Thus, the fair share distribution is as if the rate–demand of a gateway to a piconet were the minimum of the SPF and the PPF for that piconet. We call this value “*actual rate demand*” for the master/gateway pair.

**TAB. V.1 ACTUAL TRAFFIC RATE, PPF, SPF AND FAIR SHARE OF THE GATEWAY IN THE TWO PICONETS**

	piconet A	piconet B
<b>Rate demand</b>	0.7	0.6
<b>PPF</b>	0.6	0.4
<b>SPF</b>	0.5	0.5
<b>Fair share</b>	0.6	0.4



An example of this case is given in Tab. V.1, which shows the actual traffic rate, PPF, SPF and fair share of the gateway in the two piconets of Fig. V.4. In piconet B, the gateway gets a presence fraction of 0.4 (since PPF = 0.4), which is less than the SPF. In piconet A, the gateway has a PPF of 0.6 and an SPF of 0.5. Thus, the extra scatternet presence fraction of the gateway in piconet B (the difference between the SPF and the PPF) is given to piconet A, which has a higher traffic rate than may be allowed by the

SPF. Thus, the actual fraction assigned to the gateway, called the “fair share”, is 0.6 in piconet A and 0.4 in piconet B. The total fraction of the time that the gateway spends in being polled is  $0.6 + 0.4 = 1$ . We call this fraction the “total presence fraction” of the gateway. Note that the “total presence fraction” of a gateway is always  $\leq 1$ .

**TAB. V.2 RATE DEMAND, PPF & SPF FOR THE TWO GATEWAYS IN THE SCATTERNET OF FIG. V.5**

$G_1$	$M_A$	$M_B$	$M_C$	$G_2$	$M_B$	$M_D$	$M_E$
Rate Demand $G(G_1, \cdot)$	0.4	0.6	0.1	Rate Demand $G(G_2, \cdot)$	0.7	0.2	0.4
PPF	0.25	0.5	0.1	PPF	0.5	0.2	0.4
SPF	0.4	0.5	0.1	SPF	0.4	0.2	0.4
<b>Fair share</b>	<b>0.25</b>	<b>0.6</b>	<b>0.1</b>	<b>Fair share</b>	<b>0.4</b>	<b>0.2</b>	<b>0.4</b>

The situation becomes slightly more complicated when the scatternet contains more than one gateway. Consider, for example, the situation depicted in Fig. V.5, where the rate demand of each unit is shown on the arrow representing the connection link.

Tab. V.2 shows the values of PPF and SPF for the two gateways. These values have been obtained considering the original rate demand of each unit. We observe that  $G_1$  cannot get more than its PPF (0.25) from  $M_A$ , unless hurting the other units in the piconet A with a rate demand not greater than 0.25. Thus, the actual rate demand between  $G_1$  and  $M_A$  is limited to 0.25. Therefore, the extra scatternet presence fraction can be given to  $M_B$ .

In a similar way, the actual rate demand between  $G_2$  and  $M_B$  is limited to 0.4. Even though  $M_B$  might give  $G_2$  an higher rate without penalizing the other units in the piconet,  $G_2$  cannot get more than its SPF unless violating fairness with other piconets it belongs to for which the rate demand is not greater than 0.4 (see Definition V.2). Therefore, the actual rate demand between  $G_2$  and  $M_B$  is 0.4. The extra piconet presence fraction of  $G_2$  in  $M_B$  can then be given to the other units, i.e. to  $G_1$ .

```

Function maxmin_fair_scatternet(G)

% compute fair share (and PPF values) in each piconet
for j=1 to x do { R(M(j),Mj) = maxmin_fair(G,M(j),Mj) }

% compute fair share (and SPF values) in each virtual piconet
for i=1 to  $\tilde{x}$  do { R( $\tilde{M}_i, \tilde{\mathbf{M}}(i)$ ) = maxmin_fair(G,  $\tilde{\mathbf{M}}(i)$ ,  $\tilde{M}_i$ ) }

% Extract the PPF and SPF values for each gateway/master pair
(PPF,SPF) = extract_PPF_SPF(R);

% Recursion ends when PPF = SPF for each gateway/master pair
if  $\forall (\tilde{M}_i, M_j)$  PPF( $\tilde{M}_i, M_j$ ) == SPF( $\tilde{M}_i, M_j$ ) then { return R( $\cdot$ ) }

% otherwise, iterate the computation
else {
    find set MM of ( $\tilde{M}_i, M_j$ ) for which PPF( $\tilde{M}_i, M_j$ ) != SPF( $\tilde{M}_i, M_j$ );
    Extract ( $\tilde{M}_i, M_j$ )  $\in$  MM such that
        
$$\min \{ \text{PPF}(\tilde{M}_i, M_j), \text{SPF}(\tilde{M}_i, M_j) \} = \min_{(u,v) \in \mathbf{MM}} \{ \min \{ \text{PPF}(u,v), \text{SPF}(u,v) \} \};$$


    % if the tightest constraint is the PPF we fix the rate
    % of the units in the piconet with demand higher than PPF

    if PPF( $\tilde{M}_i, M_j$ ) < SPF( $\tilde{M}_i, M_j$ ) then {
         $\forall u \in \mathbf{M}(j)$  such that  $G(u, M_j) \geq \text{PPF}(\tilde{M}_i, M_j)$ , set  $G(u, M_j) = \text{PPF}(\tilde{M}_i, M_j)$  }

    % if the tightest constraint is the SPF we fix the rate of the
    % units in the virtual piconet with demand higher than SPF

    if SPF( $\tilde{M}_i, M_j$ )  $\leq$  PPF( $\tilde{M}_i, M_j$ ) then {
         $\forall u \in \tilde{\mathbf{M}}(i)$  such that  $G(\tilde{M}_i, u) > \text{SPF}(\tilde{M}_i, M_j)$ , set  $G(\tilde{M}_i, u) = \text{SPF}(\tilde{M}_i, M_j)$ ; }
    }

% Iterate with the new rate demand matrix
maxmin_fair_scatternet(G)
}

```

Fig. V.6 Pseudo-code for the computation of max-min fair share in scatternets

Consequently, the fair share distribution is as shown in the last row of Tab. V.2. We observe that the actual rate demand between a master/gateway pair coincides with the final max–min fair rate for that pair. Therefore, considering the *actual* rate demands for the gateways, instead of the original rate demand, together with the rate demand of slave units, we can compute the max–min fair rate distribution in the scatternet. A simple algorithm that can be used to compute such fair rate distribution is given in Fig. V.6.

A time–division of the gateway based on PPF and SPF takes into account that the gateway may have different traffic to different piconets. The PPF and SPF are, essentially, a measure of the amount of data the gateway can exchange with a piconet in a fair manner. Thus, such a division also removes the limitation of the Rendezvous–Point scheme, where traffic dynamics have no bearing on the time–division of the gateway.

In the next subsection, we introduce and describe an algorithm that aims to achieve such a fair distribution of bandwidth among the gateway and non–gateway slaves in a scatternet.

## **V.5. Description of algorithm**

We now describe the algorithm and show how it tends to achieve the fairness described in the previous section. We first explain the working of the algorithm in the case of a single piconet with no gateway and prove its fairness. We then extend the algorithm to the case of a scatternet and explain how the coordination between the master and the gateway is achieved.

### **V.5.A. Single piconet with no gateways**

Consider a piconet in which none of the slaves is a gateway. The master tries to calculate the traffic rate between the master and each slave. This traffic rate is the sum of the traffic rates from the master to a slave and in the reverse direction. In this section, we will assume that traffic flows only from slaves to master; masters generate no traffic to slaves. We make this assumption in order to simplify the explanation of the algorithm. The same algorithm also applies when traffic flows in both directions, from

master to slave and in the reverse direction; in this case, though, traffic rate (or other parameters) from master to slave would be replaced by sum of the traffic rates (or other parameters) in both directions.

The master uses a Round Robin polling scheme. It visits the slaves according to their Round Robin ordering, but polls a slave only if it belongs to its polling cycle. The slaves are moved in and out of the polling cycle on the basis of two variables the master maintains for each slave. These two variables are:

- $r$  – estimate of the rate of traffic generated by the slave;
- $N$  – estimate of the queue length of the slave.

When a slave is polled, the master–slave pair gets a chance to exchange a maximum amount of data denoted by  $M$ . After each such polling phase, the master updates the values of  $N$  and  $r$  in the following manner:

For the slave just polled:

$$N = N + r\tau - x \quad (\text{V.1})$$

$$r = \begin{cases} \alpha r + \beta \frac{x}{T}; & x < M \\ \alpha r + \beta \frac{x}{T} + \delta; & x = M \end{cases} \quad (\text{V.2})$$

For other slaves:

$$N = N + r\tau, \quad (\text{V.3})$$

where  $\tau$  is the time elapsed since the last update,  $x$  is the amount of data exchanged during the poll phase,  $T$  is the total time elapsed since the last poll of the same slave,  $\alpha$  and  $\beta$  are parameters used to smooth the rate estimation;  $\delta$  is a parameter used to probe for more bandwidth. Since  $N$  is an estimate of the slave's queue length and  $r$  is an estimate of the rate at which data is generated,  $N$  is increased at the rate of  $r$ . Also, when a slave is polled,  $N$  is decreased by the amount of data exchanged. After updating of these values, the master determines the changes to be made to the polling cycle. A slave

is added or deleted from the polling cycle depending upon whether its value of  $N$  is greater or smaller than a given threshold. The master now goes to the next slave according to the Round Robin ordering of slaves. If the slave is present in the polling cycle, it is polled. Else, the procedure is repeated for the next slave in the Round Robin ordering.

The value of the threshold is the minimum amount of data that the master would like the slave to have in order to poll it. We choose a value equal to the payload of a DH5 packet for the threshold since a 5-slot Bluetooth packet incurs least overhead. Thus, the master does not insert a slave in its polling cycle till its estimate of the value of  $N$  for the slave becomes greater than the threshold. This makes the simple Round Robin polling strategy adaptive to traffic and enables it to utilize bandwidth more efficiently, particularly when slaves have different rates of traffic. In the system being discussed here (single piconet with no gateways),  $M$  is set equal to the threshold. Moreover, we consider  $\beta = 1 - \alpha$ . Also, note that if the amount of data sent by the slave  $x$  is equal to  $M$ ,  $r$  is increased by a small amount,  $\delta$ . This is basically an attempt by the slave to probe for more bandwidth if it is able to send data at the present rate. We, thus, require that the rate-estimation increment should be positive when  $x$  is equal to  $M$ , i.e.:

$$\Delta r = r_{n+1} - r_n = (1 - \alpha) \left( \frac{M}{T} - r_n \right) + \delta > 0; \quad (\text{V.4})$$

where  $r_{n+1}$  and  $r_n$  are the rate-estimations at two consecutive steps. Denoting by  $S$  the number of slaves in the piconet ( $1 \leq S \leq 7$ ), we have

$$T \leq \frac{M}{r_n} + M(S - 1). \quad (\text{V.5})$$

The first term on the right hand side of inequality (V.5) is the maximum time that it may take for the value of  $N$  to become greater than the threshold, while the second term is the maximum time a slave may have to wait to get polled after getting into the polling cycle. Considering the worst case, which occurs at the maximum value of  $T$  and  $r$ , (V.5) leads to the following inequality:

$$\delta > (1-\alpha)\left(\frac{S-1}{S}\right). \quad (\text{V.6})$$

We choose the value of  $\delta$  as 0.15 and the value of  $\alpha$  as 0.85; these values satisfy the above inequality for all values of  $S$ . The maximum value of  $r$  is limited to 1, since for this value the slave is always present in the polling cycle.

Another advantage of such a scheme is that it may allow the master to go into a power-saving mode if it realizes that no slave has sufficient packets to send, i.e., if  $N$  is smaller than the threshold for all slaves. Though we do not explore this option in this context, it may be useful when Bluetooth devices work in a power-constrained environment.

To improve the working of the algorithm, we add a heuristic to it. The maximum number of polling cycles that a slave may not be polled is bounded. If a slave generates a large burst of data occasionally and then does not generate any data for a long time, the value of  $r$  for the slave may be very low. This may cause the value of  $N$  for the slave to be lower than the threshold for a long time. By limiting the maximum number of cycles missed by the slave, we make sure that such a bursty behavior of the slave does not lead to its starvation. We now explain how the above algorithm works in a Scatternet.

### V.5.B. Scatternet

The working of the algorithm in a scatternet is very similar to its operation in a piconet. The gateways are included in the round robin polling ordering of the master. The criterion for polling a gateway is the same as that of a non-gateway slave, i.e., only if the value of  $N$  is greater than the threshold.

In order to divide its time among different piconets in a fair manner, the gateway performs similar calculations as described in the earlier section for the master. The gateway maintains values of  $N$  and  $r$  for each piconet it belongs to in order to determine which piconets will be part of its “listening cycle”. The gateway hops sequentially among the piconets in the listening cycle and stays in a piconet till the end of the “listening phase”. After each listening phase, the gateway updates the values of  $N$  and  $r$ .

The gateway now includes or excludes piconets from its polling cycle depending upon whether the value of  $N$  for the piconet is greater or less than the threshold. In case the value of  $N$  is less than the threshold for all piconets, the gateway may go into a power-saving mode for a time equal to the minimum of the times it will take  $N$  (for all piconets) to reach the value of the threshold.

An important issue in which a gateway differs from non-gateway slaves is that a gateway may not be present in a piconet when a master polls it. The gateway ultimately switches to this piconet, but it misses some polls of the master before doing so. Note that this causes some fraction of the piconet bandwidth to be wasted in missed polls.

Let  $d$  be the time between the moment at which the master polls the gateway the first time and the moment at which the master polls the gateway and the gateway is present in the piconet. During this time the gateway is estimated to receive  $\Delta x = rd$  data in its queue. Thus, in order to maintain the fair behavior of the algorithm, the master gives the gateway a chance to exchange an additional  $\Delta x$  amount of data.

On the basis of the duality of the roles played by the master and the gateway (as observed in Section 0), the same formulae (V.1)–(V.3) are used to update the variables  $N$  and  $r$  in a gateway (the equations for “the slave just polled” in Section 0 apply to the piconet in which the gateway just got polled, while those for “other slaves” apply to other piconets). The master also performs the same calculations for the gateways.

The meanings of variables in these equations are similar to those for the case of a slave. Thus,  $\tau$  is the time elapsed since the last update,  $x$  is the amount of data exchanged during the listening phase, and  $T$  is the total time elapsed since the last poll from the same master. Note that, in this case,  $M = \Delta x + \text{threshold}$ .

We again employ a couple of heuristics that improve the efficiency of the algorithm. When a gateway finishes its polling by a master, it checks if any piconet has not been visited for a MAX number of slots and inserts it in the listening cycle. Thus, we limit the maximum amount of time that a piconet is not visited by a gateway. The value of MAX used in our experiments is 200 slots. Also, when a master polls a gateway, it does not know if the gateway is present in its piconet. If the gateway is not present, the slots used by the master to poll the gateway are wasted. This wastage of slots can be reduced

if the first poll packet sent by the master to the gateway is a one-slot packet. If the gateway replies to this poll, the master gives it a chance to exchange the  $M$  data.

### V.5.C. Fairness of the proposed algorithm

We now prove that the above algorithm leads to a max-min fair distribution of the bandwidth of a scatternet among units. We start by proving this in the case of a piconet. In the next step, we will extend the proof to the general case of a scatternet.

#### *Fairness in a piconet*

Let us introduce the following simplified notation:

- $S$  : number of active slaves in the piconet;
- $g_i$  : rate-demand of the  $i$ -th unit;
- $\bar{\eta}_i$  : rate achieved by the  $i$ -th unit;
- $\bar{r}_i$  : rate-estimation of the  $i$ -th unit (as defined in Eq.(V.2));

where  $\bar{\eta}_i$  and  $\bar{r}_i$  are average values. The proof of fairness can be derived from the following facts:

- a. The rate-estimation of the units that do not achieve their demand tends to reach a value of 1:  $\bar{\eta}_i < g_i \Rightarrow \bar{r}_i \rightarrow 1$ ; moreover, these units are always present in the polling cycle;
- b. Units with the same rate-estimation achieve the same average rate:  
 $\bar{r}_i = \bar{r}_j \Rightarrow \bar{\eta}_i = \bar{\eta}_j$ ;
- c. Units with a higher rate-estimation achieve an average rate at least equal to that achieved by units with a lower rate-estimation:  $\bar{r}_i > \bar{r}_j \Rightarrow \bar{\eta}_i \geq \bar{\eta}_j$ .

The statement a) derives directly from the inequality (V.4). When the rate demand is not satisfied, the amount of data that is exchanged at each polling phase is always equal to the maximum allowed,  $M$ . Then, from inequality (V.4), we have  $\Delta r_n > 0$ , and the rate-estimation increases till it reaches the maximum value of 1. Moreover, the value of  $N$

for these units is always greater than the threshold; hence, they are always a part of the polling cycle.

The statements b) and c) can be justified by the fact that the presence of a unit in the polling cycle is determined by the value of the rate–estimation (as explained in subsection V.5.A of the current section) and that the units in the polling cycle are served in a round robin fashion. Each unit can achieve, at most, the rate it requires and the aggregate rate cannot exceed 1. Thus,

$$\forall i \in \{1, 2, \dots, S\}, \bar{\eta}_i \leq g_i \text{ and}$$

$$\sum_{i=1}^S \bar{\eta}_i \leq \min \left\{ 1, \sum_{i=1}^S g_i \right\}.$$

Without loss of generality, let us partition the units into two sets, S1 and S2, in such a way that units in S1 achieve, on an average, their rate–demands, while units in S2 do not.

In case the set S2 is empty, than all the units achieve their rate–demand and the system is fair. If the set S2 is not empty, then due to the statement a), all units in S2 have a rate–estimation equal to 1. Thus, on the basis of the statements b) and c), all the units in S2 achieve the same average rate  $K$ , which is always greater than or equal to the rate achieved by any unit in S1. Moreover, S2 contains at least one unit and using statement a), this unit is always present in the polling cycle. Thus, the total system capacity is utilized. Hence, it is not possible to increase the rate of a unit in S2 without decreasing the rate of some other unit with a lower rate demand.

### ***Fairness in a scatternet***

The algorithm used by the gateway to determine its fraction of presence in different piconets is the same as that used by the master of a piconet to divide its polling among slaves. On the basis of the duality we observed in Section 0, we can apply the same argument used to prove the fairness in a piconet to prove the fairness in a “virtual piconet”. Thus, given the rate–demand of the “virtual slaves”, the gateway realizes the fair distribution of its time in the same way as described in the previous subsection.

In this case, however, the rate–demand of a “virtual slave” can be different from the real rate–demand of the gateway for the corresponding piconet. In fact, as discussed in the Section V.4.C, the gateway tries to distribute its time among the virtual slaves according to the values of SPF. But, whenever  $SPF > PPF$ , the gateway cannot get more than the PPF from the piconet and the “virtual rate–demand” to this piconet becomes equal to the PPF. Thus, the “virtual rate–demand” perceived by the gateway for each piconet is, in reality, the *actual rate demand* for that piconet, as defined in Section V.4.C. The gateway thus, distributes its time on the basis of this actual rate–demand in a max–min fair manner, as proved in the previous subsection, achieving the fair share distribution.

While discussing PPF and SPF in Section V.4, we assumed “perfect coordination” between the gateway and the master, which means that the gateway enters a piconet exactly when it is its chance to get polled and leaves the piconet on finishing its polling. Thus, if the “total presence fraction” of the gateway is 1, such perfect coordination will lead to the gateway being polled by some master at all instants of time, i.e., as soon as a gateway finishes its polling in one piconet, its polling starts in another piconet. This is obviously an ideal behavior of the system. The discussion of PPF and SPF in section V.4 is also based on the assumption that the gateway can get a total bandwidth fraction equal to its “total presence fraction”.

In reality, though, the gateway may get a total bandwidth fraction less than the “total presence fraction” since there may not be perfect coordination between the gateway and the master. The algorithm, in fact, achieves close to a perfect coordination between the master and the gateway, i.e. the gateway is given a total bandwidth fraction close in value to the “total presence fraction”. We verify this in the experiments section.

## V.6. Experiments and results

In this section, we present simulation results, which show that the algorithm satisfies the fairness criteria described earlier. We start with simple topologies that illustrate the behavior of the algorithm and then show that it also works well in more complex topologies. There are three topologies that the experiments focus on and these explain

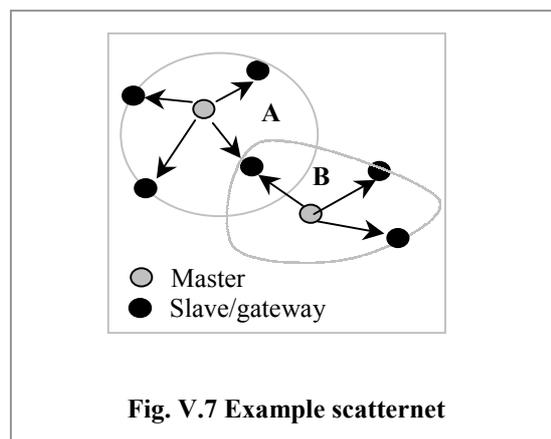
the behavior of the algorithm – a topology with a) a gateway part of two piconets, b) a gateway part of three piconets and c) a piconet having two gateways.

In the experiments, we specify the “rate of a slave”, which refers to the sum of the rates at which a slave generates data for a master and the master generates data for a slave. Moreover, unless mentioned otherwise, we assume that the traffic rate from the slave to the master is equal to that from the master to the slave. Thus, a slave having a rate of 0.4 means that the slave generates data at the rate of 0.2 Bluetooth slots per slot and the master also has a rate of 0.2 towards the slave. As we show in the section on asymmetric traffic, the algorithm works well even if these two rates are not the same.

A Bluetooth simulator written in C++ is used in the experiments. The simulator models the Bluetooth baseband and L2CAP layers and enables the creation of piconets and scatternets. All traffic generated is uniform. Each experiment is run for a system time of 32 sec.

### V.6.A. Single gateway in two piconets

We first consider the simple topology shown in Fig. V.7, which consists of two piconets, numbered A and B, connected by a single gateway. We consider various cases by changing the traffic and the number of slaves in the piconets.



#### *Adaptation between gateway and slave traffic*

Each piconet has one non-gateway slave that generates very high traffic, with rate equal to 1, to the master. The gateway has equal traffic to both masters. We vary the gateway traffic to show the fair sharing of the piconet bandwidth between the gateway

and the slave. We show the results for one piconet since the two piconets are exactly symmetric. In the graphs, “BW” in the index stands for bandwidth, “GW” stands for gateway.

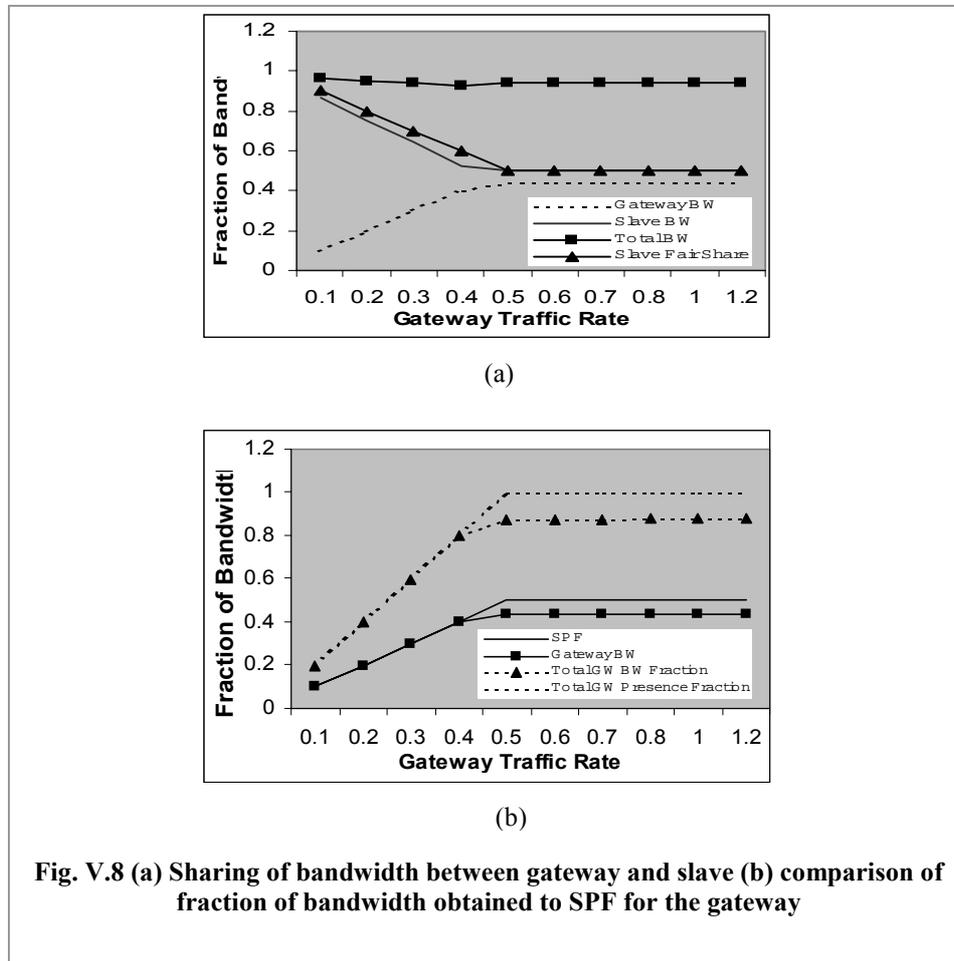


Fig. V.8(a) shows the sharing of bandwidth between the gateway and slave for different values of gateway traffic. It also shows the fair share of the slave and the total fraction of the bandwidth obtained by the gateway and the slave. In Fig. V.8(b), the comparison of the fraction of the bandwidth obtained by the gateway to its SPF (PPF and SPF are equal) is shown. Fig. V.8(b) shows that the gateway gets very nearly a fair share of the bandwidth for all values of traffic. Fig. V.8(a) shows that the bandwidth obtained by the slave adapts to the fair value. Moreover, the sum of the bandwidths obtained by the slave and the gateway is nearly equal to 1 and very little (about 3%) of the bandwidth is wasted in polling the gateway when it is not present. Fig. V.8(b) also

shows that the total bandwidth fraction obtained by the gateway is close to its “total presence fraction”.

**Different traffic to piconets**

The same topology as in the previous case, but each slave has a traffic rate of 0.3 to the master. The gateway has a fixed traffic rate of 0.2 to the master of piconet A and variable traffic to the other master. The PPF and SPF of the gateway in the first piconet are both equal to 0.2. The traffic in piconet A does not change and the gateway and the slave get a constant fraction of 0.2 and 0.3 of the piconet bandwidth respectively.

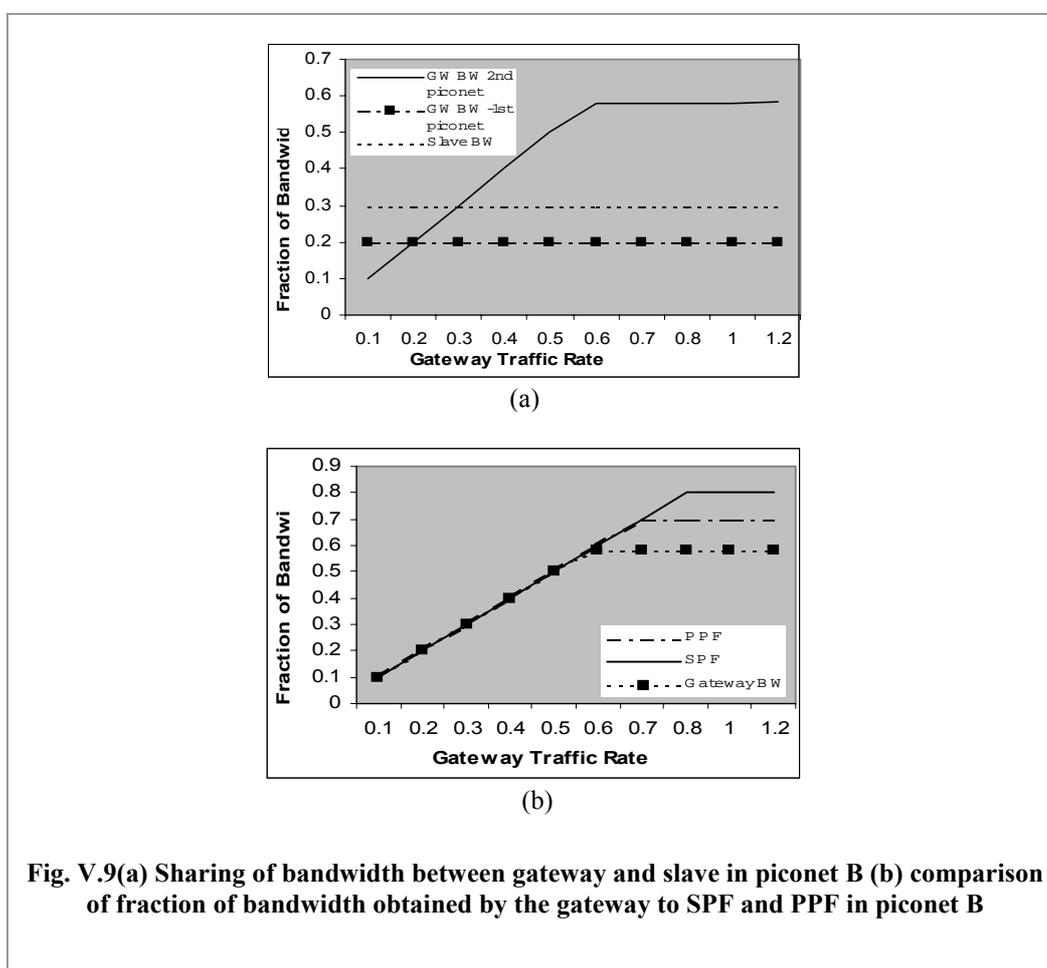
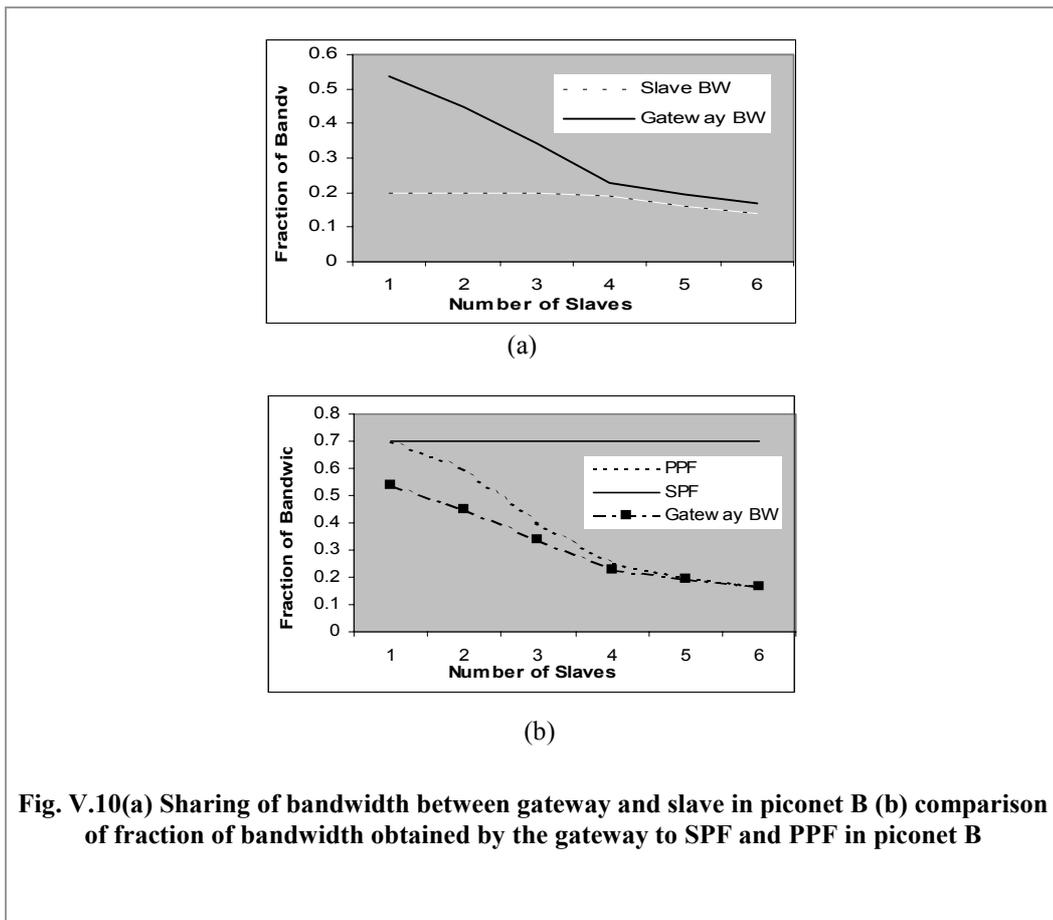


Fig. V.9(a) shows the sharing of bandwidth between the gateway and slave in piconet B for different values of gateway traffic, while Fig. V.9(b) shows the comparison of the fraction of the bandwidth obtained by the gateway in piconet B to the SPF and PPF. From the graphs, we can see that when the gateway has different traffic to piconets, it

divides its presence among the piconets according to the traffic offered and in a fair manner. Also, the gateway makes use of the lower traffic offered by the slave in piconet B to obtain a higher share of the bandwidth. Moreover, since the Gateway gets close to its fair share in each piconet, the total bandwidth fraction is close to the “total presence fraction”.

### *Different number of slaves*

Piconet A has 3 slaves, while the number of slaves in piconet B is variable. Each slave generates traffic to the master at the rate of 0.2. The gateway has a traffic rate of 0.3 to piconet A and 0.8 to piconet B. The PPF and SPF of the gateway in piconet A are, thus, 0.2 and 0.3 respectively. In piconet B, the value of PPF changes depending upon the number of slaves.



In piconet A, the slaves get a bandwidth fraction of 0.2 and the gateway gets 0.3. Fig. V.10(a) shows the sharing of bandwidth between the gateway and each slave in piconet B. Fig. V.10(b) shows the comparison of the fraction of the bandwidth obtained by the gateway in piconet B to the SPF and PPF. The gateway receives a fraction of the bandwidth approximately equal to the fair share. Also, as the number of slaves increases, the fraction of the bandwidth received by the gateway (and each slave) reduces in a fair manner.

### *Asymmetric traffic*

We now consider a case where the traffic rates from master to slave and slave to master are different (asymmetric traffic). We consider the same topology as in experiment 2), with the non-gateway slaves having the same rate as in 2). The gateway has a fixed traffic rate of 0.2 to the master of piconet A and variable traffic to the other master. The variable traffic is such that traffic from master to slave has a rate of 0.1 and traffic from slave to master varies.

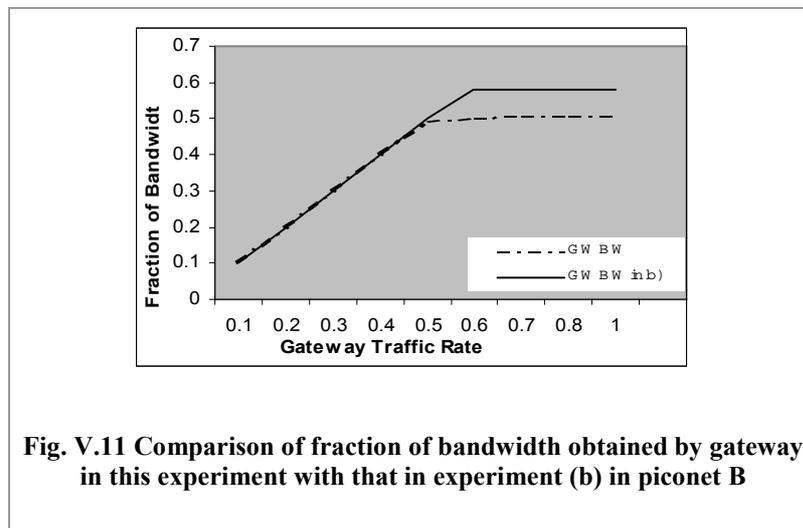
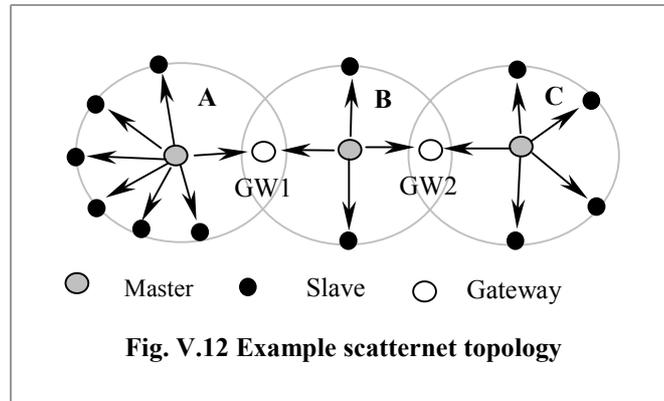


Fig. V.11 shows the comparison of bandwidth fraction obtained by the gateway versus that obtained by the gateway in experiment 2) in piconet B for different values of gateway traffic (which is the sum of Master to Slave and Slave to Master traffic rates). We see that the fraction is slightly lower than the fraction obtained in 2). Asymmetric traffic leads to wastage of slots, since an empty slot is returned in one direction where there is no data to send. It can be seen though, that the gateway still behaves in an

approximately fair manner. All other bandwidth fractions for slaves and the gateway are the same as in experiment 2).



### V.6.B. Piconet with two gateways

We now show the working of the algorithm in a piconet having 2 gateways, as shown in Fig. V.12. Piconets A, B and C have 6, 2 and 4 non-gateway slaves respectively. There are two gateways, GW 1 between piconets A and B; and GW 2 between piconets B and C. All slaves have a traffic rate of 0.2. GW 1 has a traffic rate of 0.2 in piconet A and 0.5 in piconet B. GW 2 has a traffic rate of 0.2 in piconet C. We vary the traffic rate of GW 2 in piconet B and show the fair sharing of bandwidth.

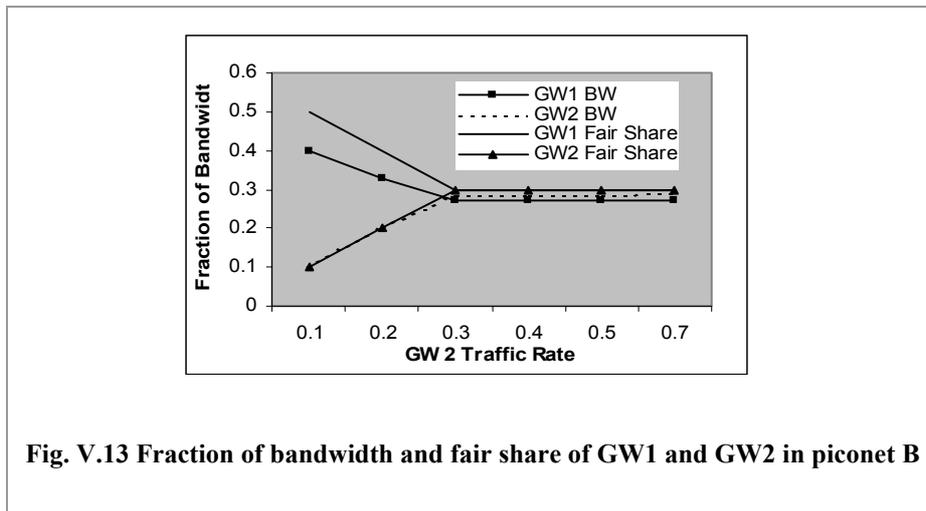


Fig. V.13 shows the fraction of bandwidth obtained by GW 1 and GW 2 in piconet B compared to their fair shares. The x-axis denotes gateway 2 traffic in piconet B. It can be seen that the bandwidth fractions obtained are very close to the fair value. Moreover,

the non-gateway slaves of piconet B receive a bandwidth fraction of 0.2, which is equal to their fair share. The bandwidth fraction received by slaves in piconets A and C does not change for different values of GW2 traffic in piconet B. The fair share of each slave (including the gateway) in piconet A is 0.1428 and the bandwidth fraction received by each slave is 0.1396; in piconet B, the fair share of each slave (including the gateway) is 0.2 and the bandwidth fraction received by each slave is 0.1975.

## **V.7. Conclusions**

In this chapter a duality principle between the roles of master and gateway in a Bluetooth scatternet has been enunciated. On the basis of such a duality, the concept of max-min fair rate distribution has been applied to a scatternet architecture. Finally, a totally distributed scatternet scheduling algorithm has been proposed. The algorithm provides an integrated solution for both intra- and inter-piconet scheduling, adapting to non-uniform and changing traffic without requiring any exchange of control information. Analysis and simulations have been used to show that the algorithm is traffic-adaptive and results in a fair allocation of bandwidth to units.

---

## VI. A HYBRID ARCHITECTURE FOR INDOOR WIRELESS COMMUNICATIONS

---

*In this chapter, a hybrid architecture of UMTS and Bluetooth is proposed. The architecture takes advantage of the complementary characteristics of these two technologies for providing a total solution for an indoor communication environment. We envision a cooperating scenario in which small Bluetooth networks (scatternets) offer basic wireless connectivity to several peripheral units that are scattered over small areas. Communication among scatternets and Internet access are, instead, provided by means of a UMTS support. We focus our analysis on a centralized topology, in which communication occurs only between peripheral units and a central access point. This topology can be applied in many different scenarios and represents an example of cooperation between third generation and PAN technologies. In addition to describing the architecture, we address the issue of fair capacity allocation in such a centralized topology and provide some analytic and simulation results for the topology considered.*

### **VI.1. Internetworking Bluetooth bubbles**

As we have indicated in Chapter I, the last few years have seen a growing demand for a global and pervasive network, which can allow people to connect anytime and anywhere. This trend has led, on the one side, to the definition of third-generation mobile telecommunication systems and, on the other side, to the development of personal area networks. The leading technologies in these two fields are the Universal Mobile Telecommunication System (UMTS) [26] and the Bluetooth [70] radio technology, respectively. While UMTS aims to provide “universal connectivity,” which means the possibility of communicating from anywhere to anywhere, Bluetooth has been proposed to provide “ubiquitous connectivity,” i.e. the possibility of communicating with every electronic device within short range.

As a matter of fact, Bluetooth is expected to be widely diffused in the near future and previsions say it will be integrated in almost every electronic device. Such a large

diffusion may promote Bluetooth-enabled devices to be used in a wide array of applications and in various network architectures. For instance, Bluetooth may provide wireless Internet Access in cafeterias, libraries, airports or, in general, in any environment in which this service is an added value offered to the customers, and low cost is more important than high performance.

However, the limited coverage range and the small number of users that can be arranged in a piconet represent severe constraints in realizing a pure-Bluetooth wireless network covering a wide area. The problem can be partially overcome by using scatternets to connect two or more piconets. Unfortunately, the performance offered by a scatternet rapidly degrades with the increasing of the scatternet size, while the complexity of maintaining and managing the structure increases. Consequently, some practical considerations limit the maximum extension of a scatternet.

The network may be extended beyond a scatternet by using other radio technologies with higher coverage range to connect scatternets. A possible solution may be based on the third generation cellular system technologies, and in particular on the unlicensed UMTS TDD [30] (Universal Mobile Telecommunication System, Time division Duplex) system, which has been specifically proposed to provide voice and data connection in indoor environments.

Each scatternet may contain a UMTS-Bluetooth hybrid unit and the UMTS TDD base station can be used to interconnect these hybrid-units in order to allow inter-scatternet communication. This leads to a hierarchical architecture, in which Bluetooth provides the basic wireless connectivity to the final users, while UMTS serves as a backbone, interconnecting several Bluetooth sub-networks. Such a solution allows exploiting the low-cost feature of Bluetooth, meanwhile overcoming its range limitations without increasing much the infrastructure costs.

This hierarchical architecture can also be implemented using some other technology, such as 802.11 [71], in place of UMTS TDD. Though a solution based on 802.11 may provide a higher bandwidth, the current 802.11b has the problem of interference with Bluetooth [19]. Moreover, mobile users will typically have a Bluetooth or UMTS interface (or both, in the future cellular phones), but not an 802.11 interface due to its high power requirements. Furthermore, since UMTS is expected to be very widely

deployed, one can imagine UMTS TDD base stations to be present in a lot of indoor environments. As a matter of fact, the UMTS base station may provide other services, like direct network access for UMTS–equipped users or local and private cordless phone service. For these reasons, we have based our hybrid architecture on UMTS TDD.

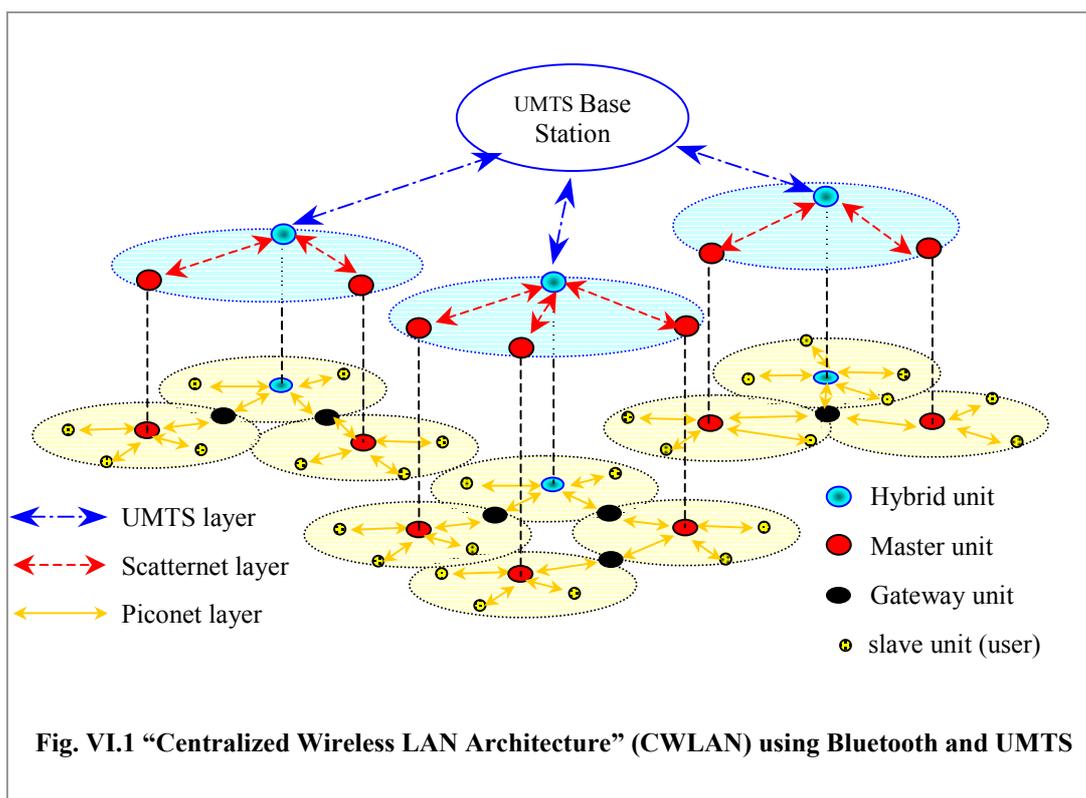
The rest of the chapter is organized as follows. Section VI.2 proposes a hybrid Bluetooth–UMTS architecture for indoor environments. Section VI.3 deals with the management of the hybrid architecture and addresses the fairness issue in such a system. Section VI.4 presents some simulation results for the proposed architecture and Section VI.5 concludes.

## **VI.2. Centralized Wireless Local Area Networks**

In this section we consider a particular topology of systems that we refer to as “Centralized Wireless Local Area Networks” (CWLANS). These are systems in which data flows only between a central node, that we call a “concentrator node” (CN), and many wireless peripheral nodes (PN) scattered in a wide area.

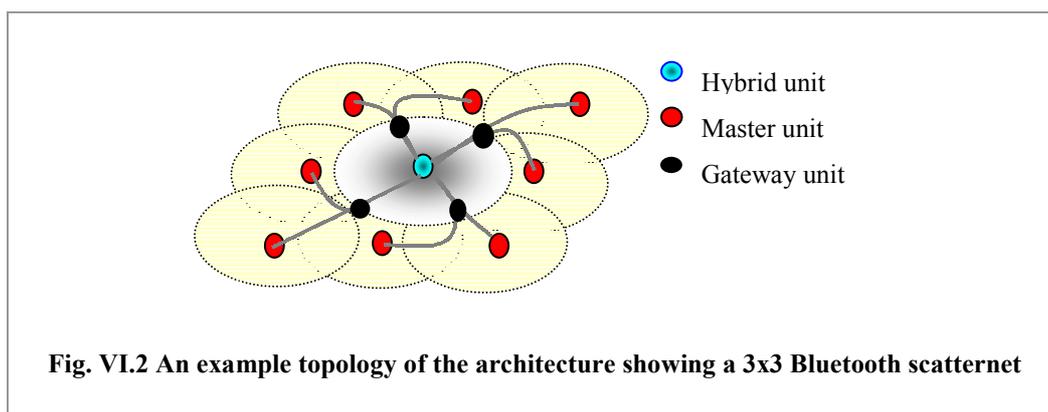
Such a topology may be applied in many different scenarios. For instance, in an “Intelligent–Supermarket,” a central server may contain information about each subscribed client, like the usual grocery list, the kind of offers he may be interested in, his account information (credit card number), and so on. When the client enters the Supermarket, his identification code is sent to the server through a wireless network. The server can, then, send back different messages that may direct the client towards the products he is interested in, or advise him about “special offers.” The messages may be displayed on the client’s cell phone or palmtop or, perhaps, on a little screen applied to the market–cart.

Another possible application for the CWLAN topology may be a cafeteria, or a library, where wireless Internet access may be offered to customers through strategically–positioned Bluetooth base stations, which may be wirelessly connected to a single Internet Access Point.



Such an architecture can be provided by a combination of Bluetooth and UMTS, using a hierarchical approach, as shown in Fig. VI.1. At the lowest level of the hierarchy, Bluetooth base stations (masters), distributed strategically in a given area, provide wireless access to users in their cells (piconets). At the second level, such piconets can be connected to form a scatternet. In each scatternet, data may be aggregated towards a Bluetooth/UMTS–hybrid unit, which serves as an interface between Bluetooth and UMTS. At the top level of the hierarchical structure, hybrid units connect, through their UMTS interface, to the UMTS TDD base station that represents the CN of the whole system.

The key element of the proposed architecture is the hybrid device. The hybrid device can operate simultaneously with both UMTS and Bluetooth interfaces. The routing layer of the hybrid device decides which interface packets are to be forwarded to. In Fig. VI.1, a packet in a scatternet may be aggregated towards the hybrid device if the destination of the packet is outside the scatternet. The packet is then forwarded to the UMTS base station (BS) from the hybrid device. The routing layer of the hybrid unit decides which interface to send the incoming packet on.



We present a specific example of this general architecture in Fig. VI.2, which shows a scatternet at the second level of the hierarchy. The Bluetooth base stations are positioned such that there are no uncovered spots. The hybrid unit is the master of the central piconet and other base stations connect to this hybrid unit through the gateways. In the figure, the gray lines connecting the hybrid unit to the other base stations show the multi-hop paths between them. Note that each gateway is shared by three piconets. The coverage area may be extended by using multiple such scatternets, which are connected to the UMTS TDD base station through their hybrid units. Clearly, in such an architecture, Bluetooth base stations, gateways, and hybrid units are static nodes: their disposition is fixed and planned in order to realize the backbone of the CWLAN.

Note that all users share the total capacity available at the CN, on the basis of the scheduling algorithm applied to the gateways. It would be desirable to give each slave an equal share of this capacity. However, users that are further away from the CN may be at a disadvantage with respect to those that are closer, since their paths to the CN may include a larger number of gateways. Moreover, a non-uniform distribution of users among piconets may lead to an unfair capacity allocation; piconets with a smaller number of users may provide higher capacity to their members than piconets with larger number of users. As we describe in the next section, a fair distribution of the bandwidth among the users may be achieved by a proper scheduling of the gateways.

### VI.3. Scatternet management

In the architecture introduced in Section VI.2, the gateways act as forwarding units. Thus, a gateway has to spend an equal amount of time in receiving data as in forwarding

it. Since a gateway can be present only in one piconet at a time, the total capacity a gateway can provide to the users it serves<sup>4</sup> is bounded by half the piconet capacity. This prevents the distribution of the capacity in a fair manner when a gateway serves more than half the total number of users in the scatternet.

We now focus on a gateway of the central piconet. If this gateway serves a fraction  $f$  of the slaves, then it spends a fraction of time equal to the minimum of  $f$  and 0.5 in the central piconet. Thus, if  $f$  is greater than 0.5, the slaves served by the gateway get less than their fair share. As explained above, the total time spent by the gateway in the other two piconets it belongs to, should be equal to that spent in the central piconet. This time is distributed between these two piconets in a fair manner, depending upon the total number of users served by each of them, where the total number of users is the sum of the users in the piconet and those served by the other gateway in the piconet.

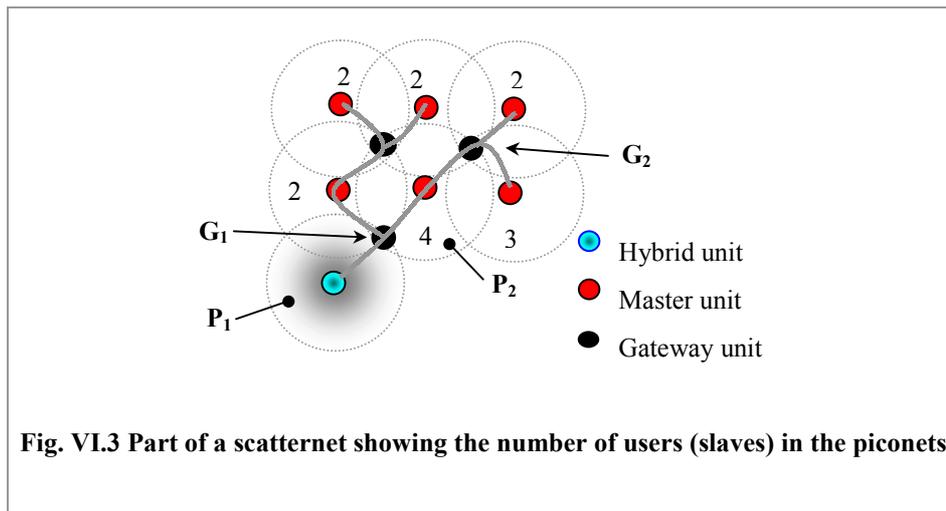
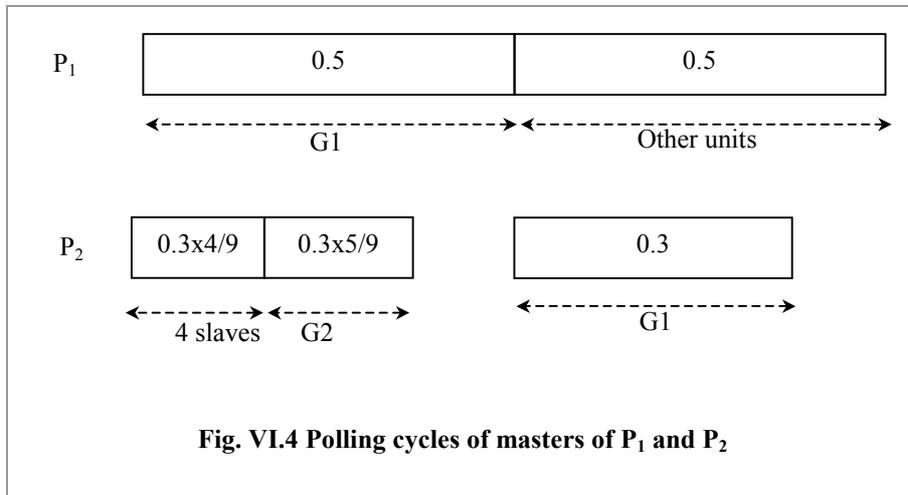


Fig. VI.3 shows an example of a part of a scatternet along with the number of users in some piconets. As before, the gray lines show the multi-hop paths between the hybrid unit and the other base stations. Note that piconets whose number of users is shown are served by the same gateway  $G_1$  of the central piconet. Let the total number of users in the scatternet be 25 (only part of them are shown). Then, the gateway  $G_1$  serves a fraction  $15/25$  of the total number of users and hence, spends a fraction 0.5 of its time in the central piconet. It divides its remaining time between the other two piconets, giving

<sup>4</sup> A user is served by a given gateway when its path to the CN includes that gateway.

each a forwarding fraction of 0.3 and 0.2 of its total time, respectively (since the piconets serve a total of 9 and 6 slaves respectively).



The forwarding fraction given to each piconet is now further divided in a fair manner. For example, the master of the piconet P<sub>2</sub> divides the forwarding fraction giving a polling fraction of 0.3 to gateway G<sub>1</sub>,  $0.3 \times 5/9$  to gateway G<sub>2</sub> and  $0.3 \times 1/9$  to each of the 4 slaves.

The system, thus, divides its bandwidth among the slaves as fairly as possible. In particular, if the number of slaves in the piconets is distributed in a uniform manner (i.e., the number of slaves served by one gateway is not greater than half of the total number of slaves), the system gives each slave an equal amount of bandwidth towards the hybrid unit.

This fair division requires a coordination of gateways and masters that may be implemented as shown by the polling cycles of masters of piconets P<sub>1</sub> and P<sub>2</sub> in Fig. VI.4. The length of the polling cycle is the same for all masters. Note that the master of piconet P<sub>2</sub> does not need to poll any unit for some time, which it may use for other activities (e.g. power saving). The gateway G<sub>2</sub> may also enter a power-save mode during the time it is not scheduled in any piconet.

If there is a change in the number of slaves in any piconet, the master of the piconet communicates it to the master of the central piconet, which reorganizes its polling cycle. This may lead to a reorganization of the polling cycles of other masters too.

The UMTS base station uses a dynamic radio resource allocation algorithm [55], where bandwidth is allocated to individual mobile terminals on the basis of their queue lengths. This can give a more efficient allocation of the UMTS channel among the hybrid units when they offer different amounts of traffic. However, if the total traffic offered to the UMTS base station exceeds the channel capacity, a hybrid unit offering higher traffic may get more than its fair share of the channel capacity. Thus, the algorithm may not guarantee fairness in such a situation.

## **VI.4. Simulation model & results**

In this section we first describe the simulation model used and then present simulation results of the proposed architecture.

### **VI.4.A. Simulation model**

We used GloMoSim [78], [88], a scalable simulation library, to develop both the Bluetooth and the UMTS models. These simulation models were then integrated into a hybrid UMTS–Bluetooth model.

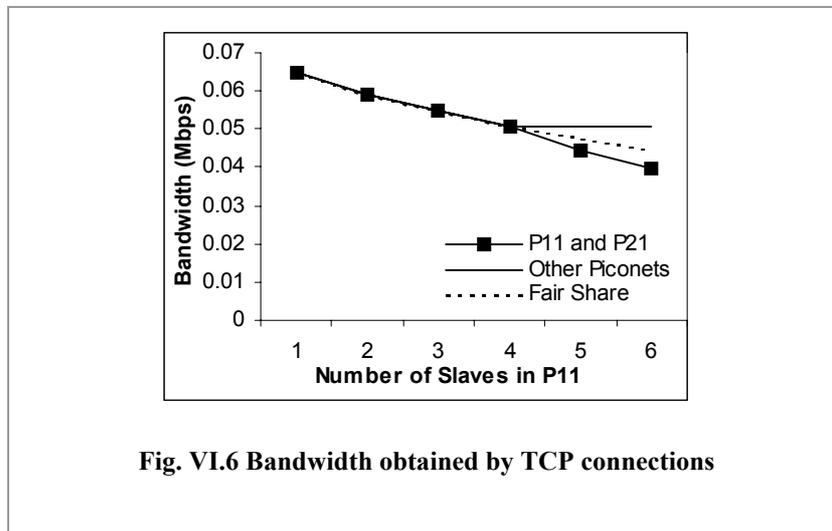
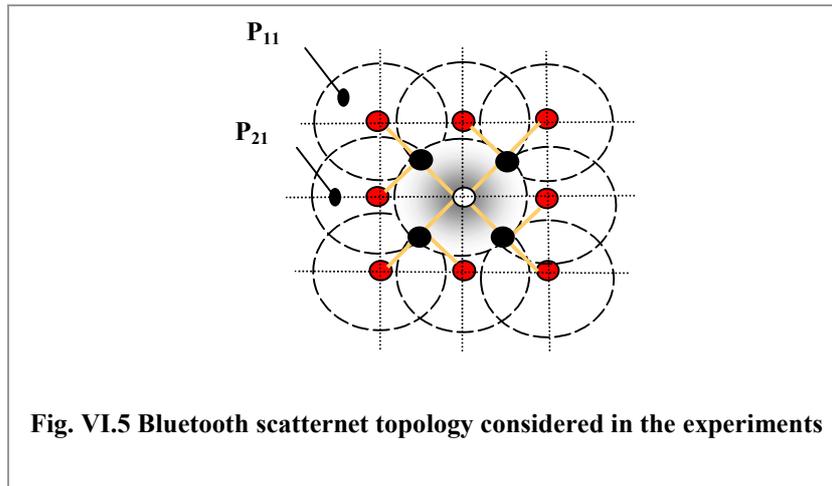
The Bluetooth simulator implements the baseband and L2CAP layers according to the specifications [70]. In the experiments, the connection type used is ACL (Asynchronous Connectionless).

The UMTS simulator was developed according to the specifications [26]. In the simulations, we adopt turbo coding with 1/3 FEC and the selective reject scheme for error control. A dynamic radio resource allocation algorithm [55] is used, as explained earlier.

The Bluetooth–UMTS hybrid model integrates both the Bluetooth and UMTS models for a comprehensive indoor communication environment. Each hybrid unit has both the Bluetooth and the UMTS interfaces and its routing layer forwards a packet on the appropriate interface.

### **VI.4.B. Simulation results**

In the experiments, the routing protocol used is the ad–hoc on demand distance vector (AODV) [61] and each experiment is run for 2 minutes of simulated time.



In the first experiment, we consider a scenario where an area of approximately 40x40 square meters is covered by a matrix of 9 piconets cell as shown in Fig. VI.5. Let  $P_{ij}$  denote the piconet at row  $i$  and column  $j$ . The hybrid unit is the master of the piconet  $P_{22}$  and connects to the UMTS TDD access point. The users are distributed such that all the piconets have 1 slave, but  $P_{21}$  that has 3 slaves and  $P_{11}$  whose number of slaves varies from 1 to 6. Each user has a TCP connection with the UMTS base station.

Fig. VI.6 shows the bandwidth obtained by a TCP connection in each piconet. When the number of slaves in  $P_{11}$  is less than or equal to 4, all slaves get a fair share of the bandwidth. When the number of slaves in  $P_{11}$  becomes greater than 4, the gateway of the central piconet, which serves  $P_{11}$  and  $P_{21}$ , serves more than half of the total number of slaves. Thus, we see in the figure that users of  $P_{11}$  and  $P_{21}$  obtain less than their fair share while other users obtain more than their fair share,.

In the next experiment, we consider three Bluetooth scatternets, each one having the same topology as shown in Fig. VI.5. In each scatternet, piconets  $P_{11}$ ,  $P_{13}$ ,  $P_{31}$  and  $P_{33}$  have one user each, while the other piconets are empty. The hybrid unit in each scatternet connects to the UMTS TDD base station. Each user performs file transfers with the UMTS base station using TCP connections. The time between two consecutive file generations is uniformly distributed. Tab. VI.1 shows file size, mean time between two consecutive file generations and total traffic offered by each scatternet (on average).

**TAB. VI.1 SIMULATION PARAMETERS**

	<b>File size (Kbit)</b>	<b>Mean separation time between files (s)</b>	<b>Total traffic (Kbit/s)</b>
<b>Scatternet I</b>	100	4	100
<b>Scatternet II</b>	100	0.8	500
<b>Scatternet III</b>	100	0.57	700

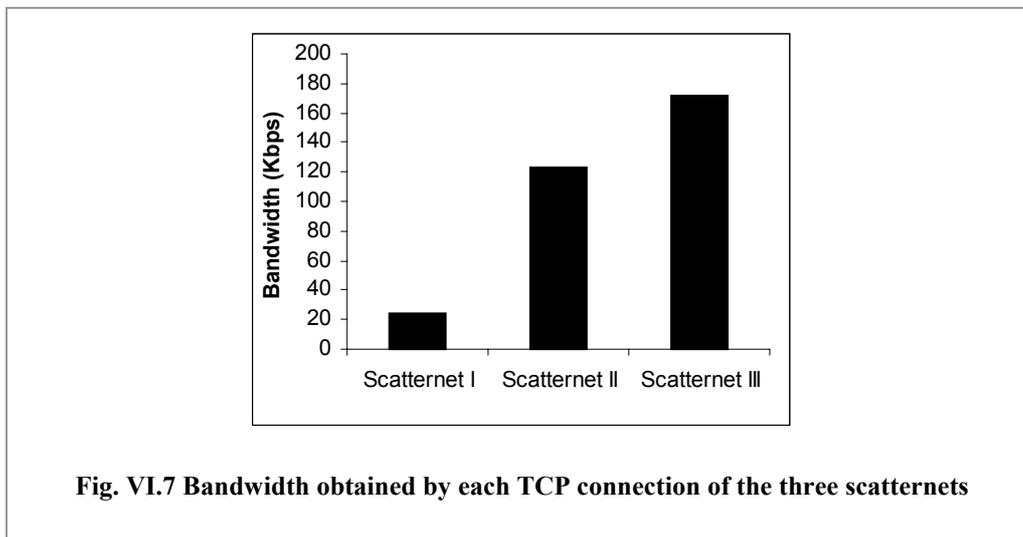


Fig. VI.7 shows the bandwidth obtained by each TCP connection of the three scatternets. It can be seen that this bandwidth is proportional to the traffic offered, due to the dynamic resource allocation algorithm used by the UMTS base station. On the contrary, a static allocation would not be adaptive to different traffic and would lead to an inefficient utilization of the UMTS bandwidth.

## **VI.5. Conclusions**

In this chapter, we have presented a hybrid architecture of UMTS and Bluetooth that can provide a solution for wireless access in an indoor environment. Such a solution has the low-cost advantage of Bluetooth without the limitations of its small range. We discussed some of the advantages of using a UMTS TDD system in this solution. We also presented a specific topology example of the hybrid architecture and addressed the issue of fairness among users through analysis and simulations. The results show that this architecture can provide fairness among the users in a scatternet as long as they are not highly concentrated in one section. Moreover, the dynamic resource allocation algorithm implemented at the UMTS base station can achieve an efficient utilization of the bandwidth.



---

## VII. MEDIUM ACCESS CONTROL IN LMDS SYSTEMS

---

*In this chapter, a hybrid access protocol known as Contention–Time Division Multiple Access (C–TDMA) is presented and analyzed in a radio cellular multi-user system scenario. C–TDMA shows some features of contention–based (Slotted–Aloha) and reservation–based (Packet Reservation Multiple Access, PRMA) protocols and it has been recommended to be used in the up-link of future European multimedia distribution systems. A simple Markov model is proposed to describe the C–TDMA behavior. To evaluate the performance of the protocol a complete statistical analysis of the model has been made. Furthermore, on the basis of the Equilibrium Point Analysis (EPA) and the C–TDMA design parameters, a fast algorithm has been developed to improve the achievable throughput of C–TDMA. Results in terms of throughput and delay under variable traffic conditions indicate that C–TDMA is able to grant optimum throughput/delay figures for typical multi–user systems. Moreover, for a digital speech scenario, a performance comparison with PRMA demonstrates that the two protocols yield substantially equivalent performance both for data and voice transmission scenario.*

### **VII.1. Introduction**

Currently, a great number of telecommunications actors (operators, research laboratories, manufacturers, standardization groups, etc.) are testing and evaluating the real market demand of interactive multimedia services for residential and business customers. As a matter of fact, trials of interactive TV and related services continue around the world [5]. Even if cable (coaxial and optical fiber) has demonstrated to satisfy the appetite for such services in most scenarios, there are particular areas where cellular radio systems offer a viable complementary solution by virtue of fast deployment, minimum infrastructure impact within cities and cost effectiveness in rural or sparse populated areas. For these reasons, an European consortium called CRABS

(Cellular Radio Access Broadband Services) has developed an ambitious project to provide digital interactive services via microwave cellular radio [23].

As regards the access protocol, CRABS has considered an efficient random access and packet-switching technique [65][67][66], easily to implement with current cellular radio technologies, to handle the up-link of the multimedia distribution systems. It has been called Contention-Time Division Multiple Access (C-TDMA).

Throughput/delay curves of C-TDMA have been obtained in the presence of a given multimedia scenario by using exhaustive computer simulations [65][67]. Numerical results show that C-TDMA is able to grant a high throughput and low access delay. On the other hand, mathematical analysis could be usefully employed to give a deeper insight into the relationship between system performance and design parameters. Unfortunately, the analysis of C-TDMA in a multimedia scenario is very complicated due to the large number of variables that have to be taken into account. Nevertheless, closed-form solutions can be obtained in a single-medium scenario by using classical mathematical tools as Markov analysis [51]. Such solution might be used to estimate the system performance in the presence of various services by dividing the global multimedia users into classes of single medium users with proportionally assigned resources. Clearly, owing to the lack of resource sharing, this simplified approach gives a conservative performance evaluation.

In this chapter, a complete statistical analysis is presented for the C-TDMA in a single-medium environment. The analysis is performed under the assumption of a Markovian model of the traffic offered to the system. However, typical values of the system parameters (number of users, number of channels, etc.), leading to a very large number of system states, cause serious difficulties to the performance evaluation through this analysis. To overcome this drawback, an effective and simple mathematical method called Equilibrium Point Analysis (EPA) [80][59] is used to obtain very significant estimates of the C-TDMA performance.

The plan of the chapter is as follows. In Section VII.2 a general description of the C-TDMA protocol is given. Section VII.3 presents the proposed statistical Markov model of the C-TDMA traffic and develops the complete statistical analysis of the system. The EPA method is applied and discussed in Section VII.4, where particular attention is

given to stability issues. Performance evaluation of C–TDMA in terms of throughput and delay are presented in Section 0. Section VII.6 introduces additional improvements of the C–TDMA in order to maximize its throughput. A performance comparison with the PRMA protocol [59] is made in Section VII.7. Finally, Section VII.8 outlines the conclusions.

## VII.2. The C–TDMA Protocol

### VII.2.A. System structure

The environment for which C–TDMA protocol has been developed is a cell in which a finite number, say  $M$ , of fixed or slowly moving users try to access some services by sharing a common radio base–station. Directional radio channels support the up–link traffic generated by each user. In order to reduce the number of collisions, the nodes use information about the availability of transmission resources, which are furnished by the base–station through a feedback broadcast channel. The time is supposed to be divided into slots of duration  $T$ . As in the conventional TDMA protocols, slots are organized in consecutive frames with  $N$  slots per frame. Messages generated by the users are fragmented into packets and each user is synchronized with the base–station in order to transmit its packets in such a way that they occupy exactly a time slot every frame.

During each frame the base–station observes the incoming traffic in order to distinguish free and reserved slots. A slot is declared to be *free* by the base–station when either it is empty or a collision among packets occurred therein. On the contrary, if a successful transmission of a single packet happened, the slot is declared *reserved*. At the end of the frame the base–station broadcasts the list of the free slots to the users.

On the basis of the free slot list, the users that at the beginning of a frame have a message to be transmitted try to occupy a free slot, according to a contention policy that will be discussed in the following. If the attempt is successful, say at the slot  $k$  of the frame, this is notified to the node by the absence of the slot  $k$  in the free slot list at the beginning of the next frame. Then, the node may continue to transmit the packets of its message in the slot  $k$  of the following frames without fear of collisions, because the slot does not appear to be free to the other nodes. On the contrary, if the attempt fails for a

collision, the slot  $k$  appears to be free in the next list, so that the node becomes aware of the failure and tries again (possibly in a different slot) in the next frame. The nodes extract the slot synchronization from the base–station feedback timing.

The C–TDMA protocol differs from the R–ALOHA [51][80] in that it does not use a broadcast up–link, so that information about the state of the slots must be furnished by the base–station. Moreover, it differs from the PRMA protocol [59][34] in that the slot state is notified by the base–station only once per frame, with a very little overhead.

Note that during the transmission of the last packet of a message the corresponding slot appears as a reserved one. Then, it is not in the free slot list for the following frame and consequently it will not be occupied by another node. To avoid this drawback, according to [51], the protocol can be improved by using an *end–of–use flag* version, in which the last packet of a message contains a special purpose flag (f.i. in the last bit), indicating to the base–station the end of the reservation, without requiring complex handling of the packet content. Then, the base–station can include the slot in the free slot list for the next frame. A trouble arises when a node transmits a message consisting of only one packet, because the successive appearance in the free list of the slot used for the transmission could be due both to a successful transmission or to a collision. Possible countermeasures involve slight modifications of the detection at the base–station and of the feedback information.

### **VII.2.B. Contention policy**

We assume as a very natural policy that the nodes apply a *random choice* among the free slots. Namely, a node in the contention state chooses at random one of the free slots in the list transmitted by the base–station and tries to occupy it, by transmitting its first packet therein. Note that, while this policy guarantees a small number of collisions under low traffic conditions, with large traffic it may cause iterated collisions. In order to reduce this phenomenon, we consider the *permission of transmission and random choice* policy. Namely, the first packet of a message is allowed to be transmitted in the next frame with probability  $p$  (on the basis of an internal pseudorandom number generator) and, only if permission is obtained, the node chooses at random the transmission slot. The *permission probability*  $p$  is a design parameter that should be

optimized for best throughput and transmission delay. Of course the pure random choice policy is a particular case (for  $p=1$ ) of the more general policy.

### VII.3. The mathematical model

#### VII.3.A. The offered traffic

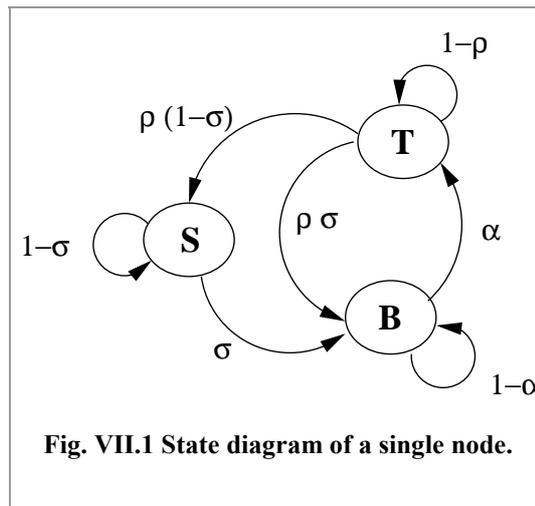
We assume that each node alternates intervals in which it is transmitting a message, and intervals in which it does not transmit and waits for a new message or contends for beginning a new transmission. The transmission intervals expressed in frames, and consequently the message lengths expressed in packets, are assumed to be independent geometrical random variables with mean  $1/\rho$ . Similarly the waiting times for new messages (expressed in frames) are considered as independent geometrical random variables with mean  $1/\sigma$ . This characterization of the traffic at a single node is assumed to be independent of both the past evolution of the network and the present situation of the other nodes.

We define the traffic offered by a node as the traffic that the node would transmit on a channel of its own, i.e.

$$g = \frac{\frac{1}{\rho}}{\frac{1}{\rho} + \frac{1}{\sigma}} = \frac{\sigma}{\rho + \sigma} \text{ packets/frame.} \quad (\text{VII.1})$$

By assuming that the nodes are uniformly loaded, the global offered traffic is

$$G = \frac{M\sigma}{\sigma + \rho}. \quad (\text{VII.2})$$



### VII.3.B. Node states

The system behavior may be modeled in the following way. Each node can be represented as a three state machine (Fig. VII.1): the *silent* state (S), in which the node has no messages to be transmitted, the *talking* state (T), in which the node is transmitting its message, the *backlog* state (B) in which the node has a new message and tries to begin its transmission according to the policy described above. In order to simplify the analysis we found convenient to consider the frame in which the node generates a new message as belonging to the backlog phase. The state transitions are supposed to occur at the beginning of every frame.

By virtue of the memoryless statistics of the message lengths, when the node is in T, either it transmits a new packet and remains in T with probability  $1-\rho$  or it ceases to transmit and leaves T with probability  $\rho$ . As a consequence of the assumption about the backlog state, being  $\sigma$  the probability that a new message is immediately generated, the transition occurs directly to the state B with probability  $\rho\sigma$  while with probability  $\rho(1-\sigma)$  the new state is S. Similarly, according to the memoryless statistics of the silence time, the node remains in S with probability  $1-\sigma$  while passes to B with probability  $\sigma$ . Finally, when the node is in B during a frame, it will pass to T in the following frame with probability  $\alpha$  if its contention procedure is successful and remains in B with probability  $1-\alpha$ .

Two facts deserve to be remarked. First, while the transitions from both T and S depend on the single node behavior, the contention phase is intimately related to the global behavior of the system and the probability  $\alpha$  depends on the present state of all the nodes of the network. Second, it is implicitly assumed that a node cannot generate new messages until the transmission of the previous message is completed. Then, the amount of data generated is dependent on the channel condition. This traffic model is suitable to describe only applications that are not too sensitive to delays, or to variations in delay. On the contrary, the traffic generated by real time applications, as digital speech or video transmission, is independent on the channel condition; consequently it requires a mathematical model different from that proposed in this chapter. In the following we consider only the first kind of applications, deferring to Section VII.7 the analysis of some results obtained in a digital speech scenario.

### VII.3.C. System variables

We assume as state variables of the system the quantities:  $b_n$ ,  $t_n$ , and  $s_n$ , namely the number of the nodes in the states B, T, and S, respectively, at the end of the  $n$ -th frame. Of course, one of the three variables is linearly dependent on the other two, because their sum must equate the total number of nodes:  $b_n + t_n + s_n = M$ . Note that the state variables  $t_n$  and  $b_n$  may assume the values  $0, \dots, N$  and  $0, \dots, M$  respectively, with the further constraint  $b_n + t_n \leq M$ . The time evolution of the state variables is governed by the following equations

$$\begin{aligned} s_{n+1} &= s_n + x_n - w_n \\ b_{n+1} &= b_n + y_n + w_n - z_n \\ t_{n+1} &= t_n + z_n - x_n - y_n \end{aligned} \tag{VII.3}$$

where

- $x_n$  is the number of nodes passing from state T to state S in the  $n$ -th frame;
- $y_n$  is the number of nodes passing from state T to state B in the  $n$ -th frame;
- $w_n$  is the number of nodes passing from state S to state B in the  $n$ -th frame;
- $z_n$  is the number of nodes passing from state B to state T in the  $n$ -th frame.

Even though one of Eqs. (VII.3) is redundant in that it is linearly related to the others, we shall take it into account also in the following, in order to get a deeper insight of some issues, particularly related to the network stability. In the following, we show that, under the assumptions made, the vector process  $w_n = (b_n, t_n)$  is a Markov chain. For this purpose we study the statistics of the random variables  $x_n, y_n, w_n, z_n$ , and prove that they depend only by the present values of  $b_n$  and  $t_n$  and are independent of their past evolution.

### **Statistics of $x_n$ and $y_n$**

The variables  $x_n$ , and  $y_n$  statistically depend only on the number of the transmitting nodes  $t_n$ , through the memoryless mechanism of both transmission and silence duration. In particular, under the condition that  $t_n = t$ ,  $y_n$  is a binomial random variable of index  $t$  and parameter  $\rho\sigma$ . Analogously,  $x_n$  is a binomial random variable of index  $t$  and parameter  $\rho(1-\sigma)$ . In particular, we get

$$\begin{aligned} P[x_n = x, y_n = y | b_n = b, t_n = t] &= \\ &= \binom{t}{x+y} \binom{x+y}{x} \rho^{x+y} (1-\rho)^{t-x-y} \sigma^y (1-\sigma)^x \end{aligned} \quad (\text{VII.4})$$

The conditional average values of  $x_n$  and  $y_n$  are given by

$$E[y_n | b_n = b, t_n = t] = \rho\sigma t, \quad (\text{VII.5})$$

$$E[x_n | b_n = b, t_n = t] = \rho(1-\sigma)t. \quad (\text{VII.6})$$

### **Statistics of $w_n$**

The statistics of  $w_n$  depend only on the number of silent nodes  $s_n = M - b_n - t_n$  and is independent of the past evolution of the network. Namely, provided that  $b_n = b$  and  $t_n = t$ ,  $w_n$  is a binomial variable with index  $M - b - t$  and parameter  $\sigma$ , so that

$$P[w_n = w | b_n = b, t_n = t] = \binom{M-b-t}{w} \sigma^w (1-\sigma)^{M-b-t-w}. \quad (\text{VII.7})$$

Moreover

$$E[w_n | b_n = b, t_n = t] = \sigma(M - b - t). \quad (\text{VII.8})$$

### *Statistics of $z_n$*

The variable  $z_n$  depends on the number of backlogged nodes  $b_n$ , on the number of free slots in the list  $N - t_n$ , and on the contention policy. Preliminarily we note that, provided that  $b_n = b$ , on the basis of the permission rule, the number of really contending nodes reduce to  $c_n$ , a binomial variable with index  $b$  and parameter  $p$ , namely

$$P[c_n = w | b_n = b, t_n = t] = \binom{b}{c} p^c (1-p)^{b-c}. \quad (\text{VII.9})$$

The somewhat cumbersome computation of  $P[z_n = z | b_n = b, t_n = t]$  is deferred to appendix VII-A. As regards the conditional expectation, the number of nodes gaining the slot reservation is given by

$$z_n = \sum_{i=1}^{c_n} \chi_i, \quad (\text{VII.10})$$

where  $\chi_i$  is a  $\{0,1\}$ -variable assuming value 1 if the  $i$ -th contending node succeeds, i.e. if no other contending node transmits in the slot chosen by the  $i$ -th node among the  $N - t_n$  available slots. Then the success probability is given by

$$P[\chi_i = 1 | c_n = c, t_n = t] = \left(1 - \frac{1}{N-t}\right)^{c-1}; \quad (\text{VII.11})$$

of course the above probability vanishes if  $t = N$  (no free slots) or if  $c = 0$  (no contending nodes).

In conclusion, for  $t < N$  we get

$$\begin{aligned}
& \mathbb{E}[z_n | b_n = b, t_n = t] = \\
& = \sum_{c=1}^b \mathbb{E}\left[\sum_{i=1}^{c_n} \mathcal{X}_i | c_n = c, b_n = b, t_n = t\right] \mathbb{P}[c_n = c | b_n = b, t_n = t] \quad (\text{VII.12}) \\
& = \sum_{c=1}^b c \left(1 - \frac{1}{N-t}\right)^{c-1} \binom{b}{c} p^c (1-p)^{b-c} = bp \left(1 - \frac{p}{N-t}\right)^{b-1}
\end{aligned}$$

while  $\mathbb{E}[z_n | b_n = b, t_n = N] = 0$ . When the end-of-use flag approach is followed, the random choice occur among  $N - t_n + x_n + y_n$  slots, including the slots just released by the nodes ending to transmit.

As a conclusion, the vector process  $\omega_n = (b_n, t_n)$  is a Markov chain, whose transition probabilities

$$\begin{aligned}
& \mathbb{P}[\omega_{n+1} = (b', t') | \omega_n = (b, t)] = \\
& = \mathbb{P}[y_n + w_n - z_n = b' - b, z_n - x_n - y_n = t' | b_n = b, t_n = t] \quad (\text{VII.13})
\end{aligned}$$

can be trivially computed by (VII.4), (VII.7), and (VII-A.8), taking into account that the bivariate  $(x_n, y_n)$  and the variables  $w_n$  and  $z_n$  are statistically independent under the condition  $b_n = b, t_n = t$ . For reasonable values of  $M$  and  $N$  the number of states of the Markov chain (approximately  $NM$ ) is too large to allow standard applications of the Markov analysis. Consequently, the performance of the protocol has been derived with a different approach. However, we have computed and used the transition probabilities in order to validate the results discussed in the following for relatively little values of these quantities.

## VII.4. Equilibrium Point Analysis

The analysis of the performance of the C-TDMA protocol will be conducted by using the Equilibrium Point Analysis (EPA), which was introduced by Tasaka [80] for the R-ALOHA protocol and subsequently used by Nanda *et al.* [59] [58] for the PRMA protocol. The same approach had been previously applied by Carleial and Hellman [15]

and by Kleinrock and Lam [48] in their pioneering analysis of the bistability of the ALOHA protocols.

The approach, particularized to the present case, is based on the following considerations. If the present state assumes an assigned value  $\omega_n = (b, t)$ , the optimal mean square error estimation  $\hat{\omega}_{n+1}$  of the next state is given by

$$\begin{aligned}\hat{b}_{n+1} &= E[b_{n+1} | b_n = b, t_n = t], \\ \hat{t}_{n+1} &= E[t_{n+1} | b_n = b, t_n = t].\end{aligned}\tag{VII.14}$$

Now, we are induced to consider as *equilibrium points* of the system the states  $(b, t)$  such that  $\hat{\omega}_{n+1} = \omega_n = (b, t)$ , i.e. the solutions of the equations

$$\begin{aligned}b &= E[b_{n+1} | b_n = b, t_n = t], \\ t &= E[t_{n+1} | b_n = b, t_n = t].\end{aligned}\tag{VII.15}$$

Of course, such solutions are not integer, at least in general. However, equations (VII.15) can be interpreted as *equilibrium curves* dividing the state space into regions where the differences  $\hat{b}_{n+1} - b$ ,  $\hat{t}_{n+1} - t$  have constant signs.

The derivation of equations (VII.15) on the basis of the statistics of the previous section is straightforward. Indeed, by applying to (VII.3) the conditioned averages found above, (VII.15) become

$$bp \left(1 - \frac{p}{N-t}\right)^{b-1} = \sigma(M - b - t) + \rho\sigma t\tag{VII.16}$$

$$bp \left(1 - \frac{p}{N-t}\right)^{b-1} = \rho t\tag{VII.17}$$

The above equations can be obtained by equating the average input and output flows for nodes B e T of Fig. VII.1, with

$$\alpha = p \left( 1 - \frac{p}{N-t} \right)^{b-1} \quad (\text{VII.18})$$

Analogously, the flow equilibrium equation of the node S is

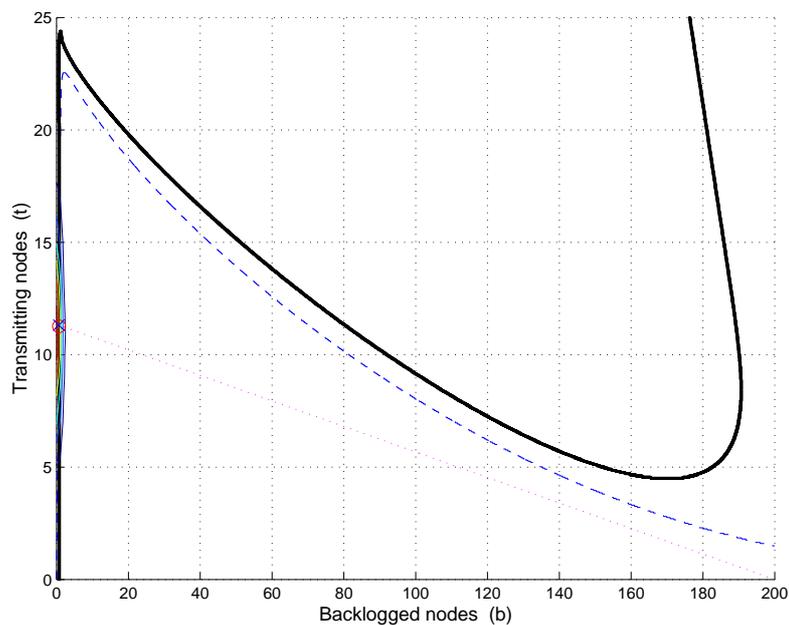
$$\rho(1-\sigma)t = \sigma(M-b-t) \quad (\text{VII.19})$$

linearly related to equations (VII.16) and (VII.17). Finally, because these equations are meaningless for  $0 < b < 1$  and  $N-p < t < N$ , we have extended them to this region by linear interpolation. Preliminarily we note that the system formed by (VII.16), (VII.17), and (VII.19) admits solutions. As a matter of fact, the line (VII.19) crosses the axes  $b$  and  $t$  in the points  $(M, 0)$  and  $(0, \sigma M / (\rho + \sigma - \rho\sigma))$ , while (VII.17) represents a curve passing through the origin and the point with coordinates  $(M, t_M)$ , where  $t_M$  is the solution of the equation

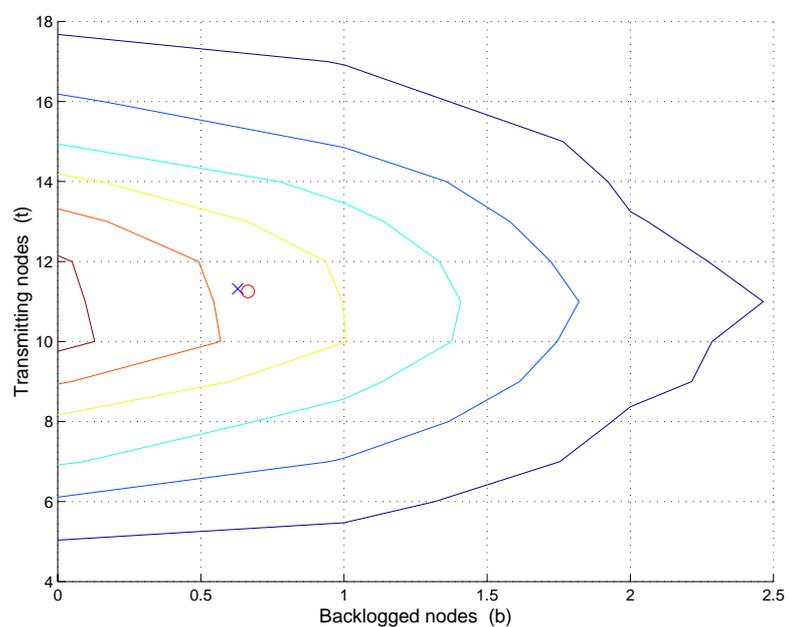
$$Mp \left( 1 - \frac{p}{N-t_M} \right)^{M-1} = \rho t_M; \quad (\text{VII.20})$$

this solution turns out to be positive, as it may be verified by simple analytical considerations. Owing to the continuity of the curves in the interval  $[0, M]$  we may expect an odd number of crossing points.

Fig. VII.2 (a) shows the three equilibrium curves for particular values of the system parameters. The dashed one is the flow equilibrium curve of the node T (see (VII.17)), while the solid curve is the equilibrium curve of the node B (see (VII.16)). Finally, the dotted line is the equilibrium curve of the node S. The curves exhibit a single common equilibrium point on the increasing side of the T equilibrium curve. Note that in this point the number of transmitting nodes increases with the number of contending ones. The level curves around the equilibrium point, sketched with greater detail in Fig. VII.2 (b), represent the stationary probability distribution of the state  $(b_n, t_n)$  computed by a simulation program. As the figure shows, the equilibrium point (denoted by the cross) is quite near to the probability distribution average, denoted by the circle. The results show the substantial accuracy of the equilibrium point approach.

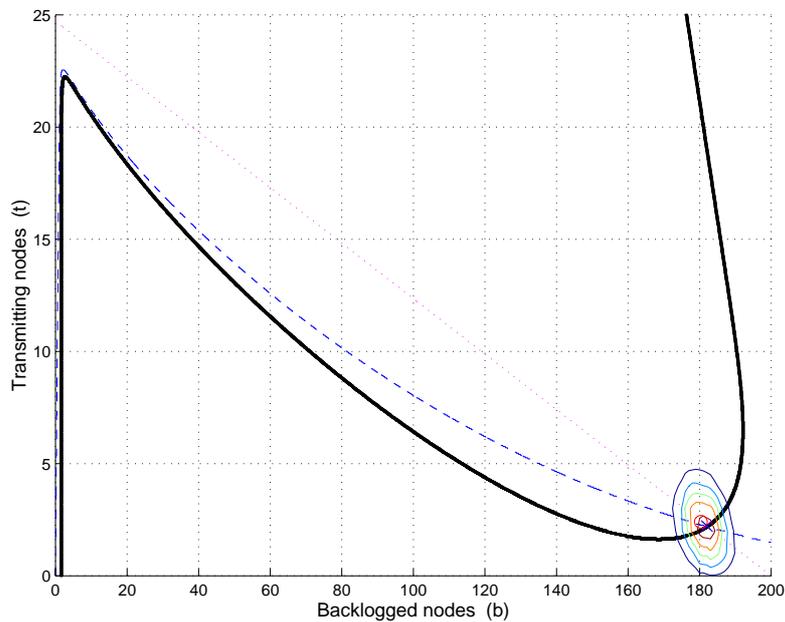


(a)

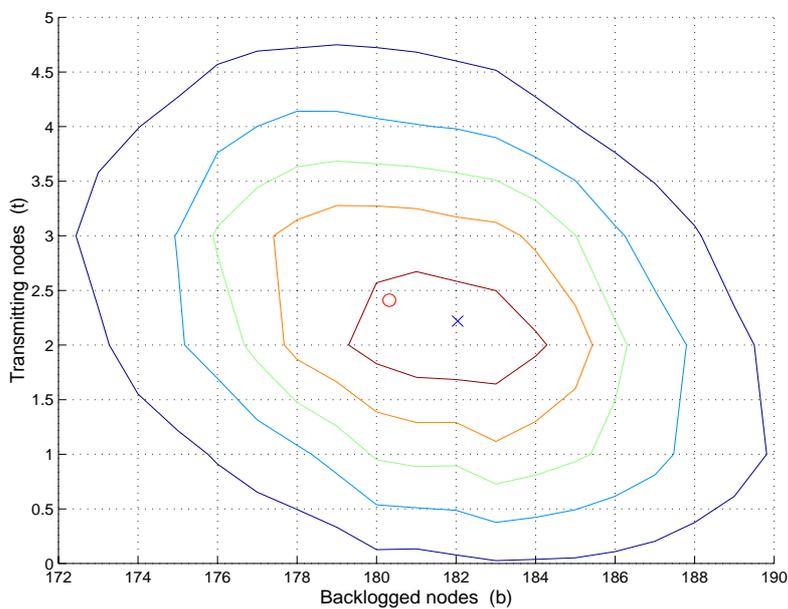


(b)

**Fig. VII.2** Equilibrium curves (a) and expansion of level curves of the stationary distribution (b).  
 ( $M=200$ ,  $N=25$ ,  $\rho=0.05$ ,  $\sigma=0.003$ ,  $\$$  and  $p=0.9$ ).



(a)

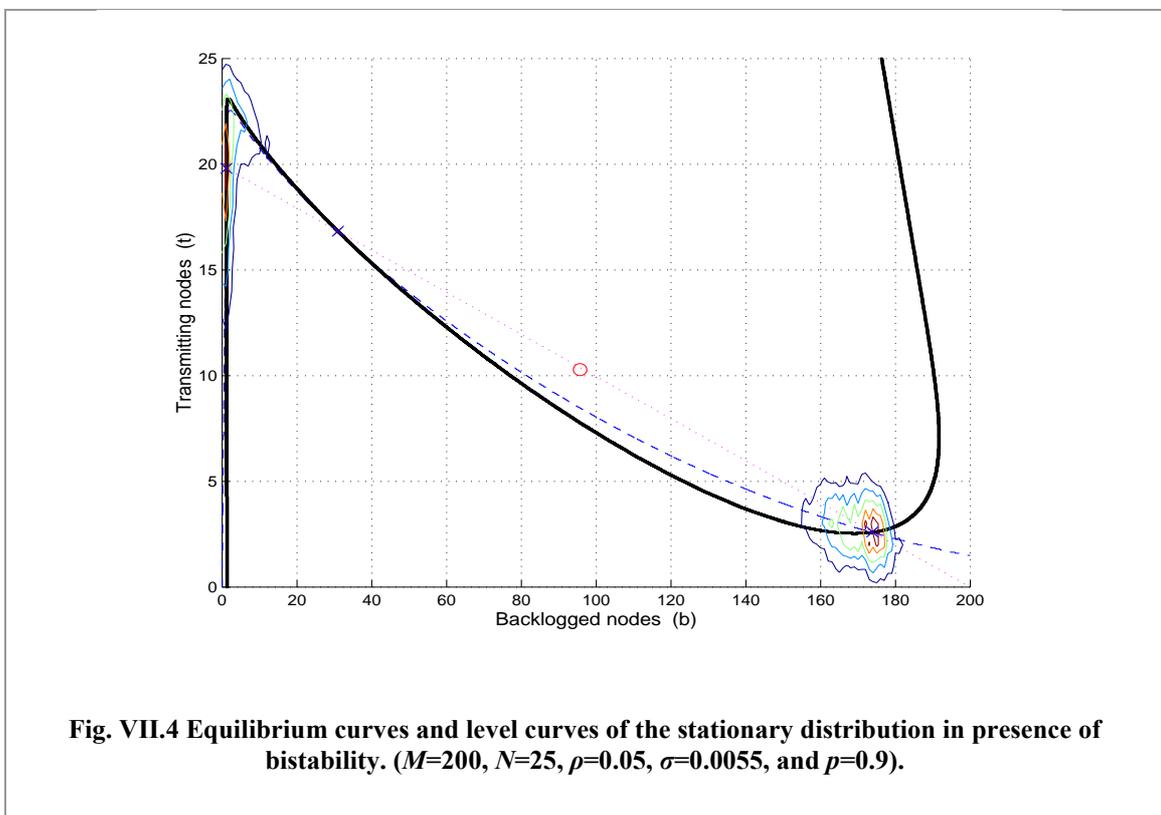


(b)

**Fig. VII.3** Equilibrium curves (a) and expansion of level curves of the stationary distribution (b).  
 ( $M=200$ ,  $N=25$ ,  $\rho=0.05$ ,  $\sigma=0.003$ ,  $\$$  and  $p=0.9$ ).

Fig. VII.3 shows the equilibrium curves (with different values of the system parameters) with a single equilibrium point in the side of the T equilibrium curve where the transmitting nodes decrease when the contending nodes increase. Also in this case the equilibrium point and the average state are practically coincident.

Finally we note that the solution of the equilibrium equations shows that the C-TDMA protocol is affected by the typical bistability phenomena appearing in the ALOHA and ALOHA derived protocols [80][59][15][48]. Fig. VII.4 shows a case in which the equilibrium curves have three different common points. The simulated distribution probability is substantially bimodal and two of the equilibrium points are centered into clusters of frequently visited states, whereas the intermediate equilibrium point belong to a rarely visited region. The simulation shows that the Markov chain tends to alternate periods of permanence in the two clusters and that the first exit time from one cluster is approximately proportional to the inverse of the distance from the intermediate equilibrium point. Note that the average state is different from each of the equilibrium points, so that to consider it as a relevant index of the system behavior is completely misleading.



## VII.5. Throughput and Delay

The throughput and the delay can be obtained from the results of the EPA, provided that the computation approach gives a single equilibrium point. As regards the throughput, the number of packets transmitted during a frame coincides with the number of nodes in the transmission state  $T$  in the same frame. Then the equilibrium useful traffic expressed in packets per frame is given by

$$S = t \quad (\text{VII.21})$$

Of course  $S \leq N$  and we can define the *channel utilization* as  $\gamma = S/N = t/N$ . Moreover we define the *efficiency* of the protocol as

$$\eta = \frac{S}{G} = \frac{t(\rho + \sigma)}{M\sigma}. \quad (\text{VII.22})$$

As regards the *contention delay*, i.e. the number  $\tau$  of frames that a node spends in waiting to reserve a slot, we make the following argumentation. For each node the time can be subdivided into cycles formed by a silent period, with mean duration  $1/\sigma$ , followed by a (possibly missing) contention period with mean  $\tau$ , followed in turn by a transmission period with mean  $1/\rho$ . Then the equilibrium traffic of the system is

$$S = M \frac{\frac{1}{\rho}}{\frac{1}{\sigma} + \tau + \frac{1}{\rho}}. \quad (\text{VII.23})$$

Incidentally, the equation shows the mechanism used by the system to maintain its stability even though the offered traffic  $G$  overwhelms the system capacity (i.e.  $N$  packets/frame). Indeed the system reacts to overloads by increasing the lengths of the cycles and consequently by deferring the generation of new messages. This behavior is slightly different by that of the PRMA protocol, which reacts to overloads by discarding packets.

Eqs. (VII.21) and (VII.23) lead to the following expression of the contention delay (in frames)

$$\tau = \frac{\sigma M - (\rho + \sigma)t}{\sigma \rho t} = \frac{1}{\rho} \frac{G - S}{gS}. \quad (\text{VII.24})$$

Now, we discuss the results obtained in terms of different parameters as the single user offered traffic  $g$ , the number of users  $M$ , the permission probability  $p$ . The number of slots in a frame is maintained at a fixed value  $N=25$ .

Fig. VII.5 shows the throughput and the delay as a function of the users number for  $g=0.4$ ,  $p=0.9$  and different values of the parameter  $\rho$ . For low values of  $\rho$ , i.e. for a high average length of the message, the throughput increases linearly with the number of users and reaches a maximum value quite near to the channel capacity, with approximately  $M = N/g$ . For a greater number of users the throughput gradually decreases, while the delay increases very rapidly. This can be explained by considering that in the saturated channel, a new message must wait the end of a transmitted message: then, if the messages are long, also the waiting time is long. For high values of  $\rho$ , i.e. for short messages, the linear increase of the throughput is limited to a lower number of users, with reduced efficiency. This can be attributed to the fact that frequent attempts of transmission induce a growth of the collisions number.

Fig. VII.6 shows throughput and delay with  $p=0.2$ , i.e. with a reduced permission probability. The comparison with the previous figure shows that the degradation of the performance above the saturation is reduced. On the other hand, below the saturation the efficiency of short messages traffic is reduced and the delay increases. The growth of the delay can be explained by noting that, also in the absence of offered traffic, a single message is affected by a delay of  $1/p$  frames introduced by the permission mechanism. The above results have been confirmed by a large amount of computer simulations.

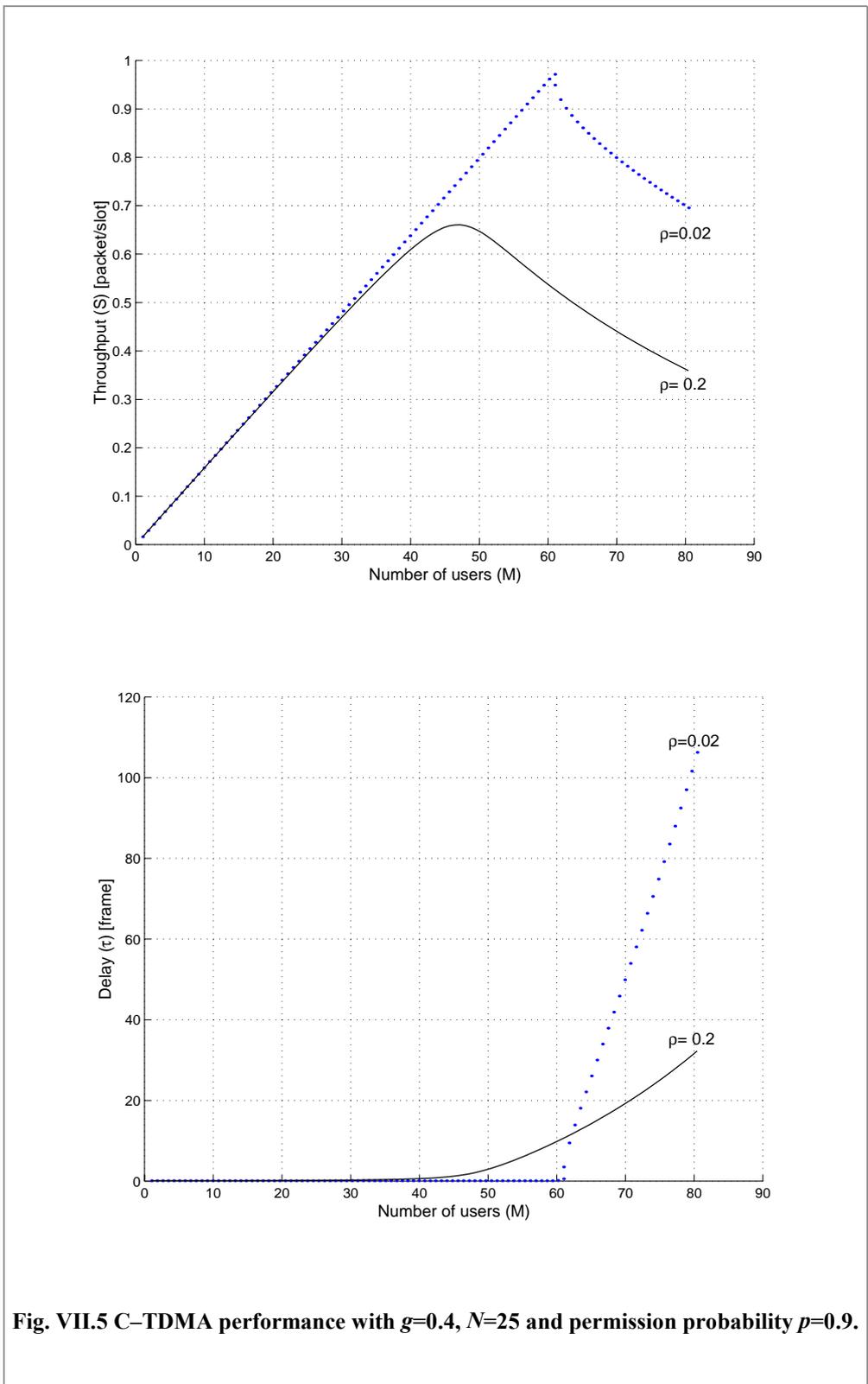


Fig. VII.5 C-TDMA performance with  $g=0.4$ ,  $N=25$  and permission probability  $p=0.9$ .

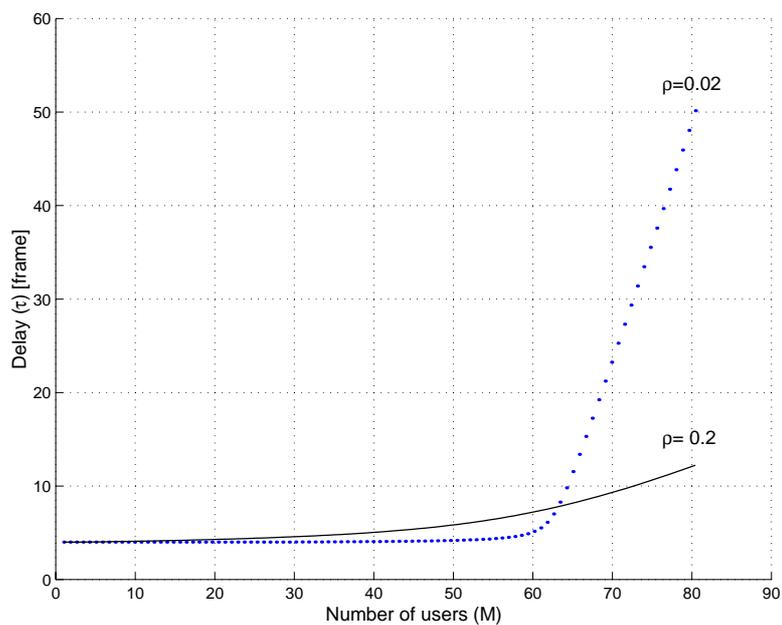
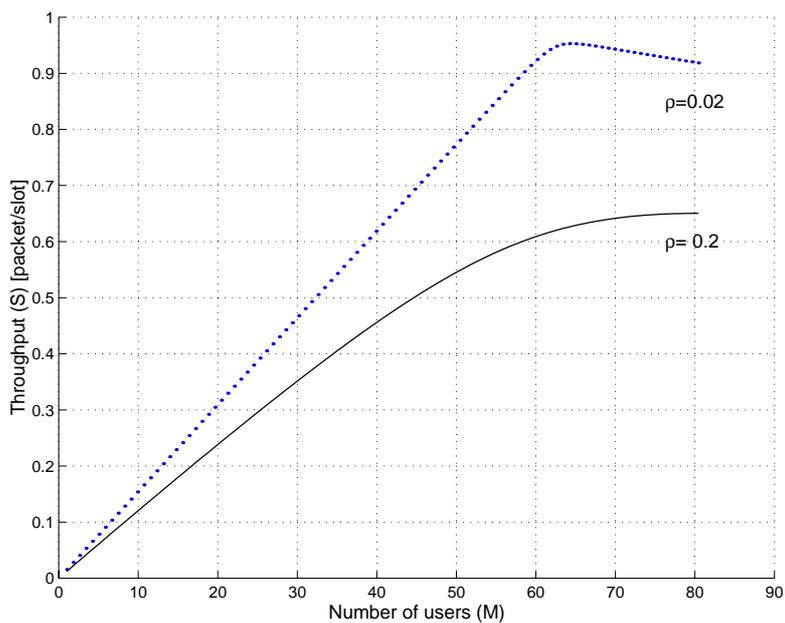
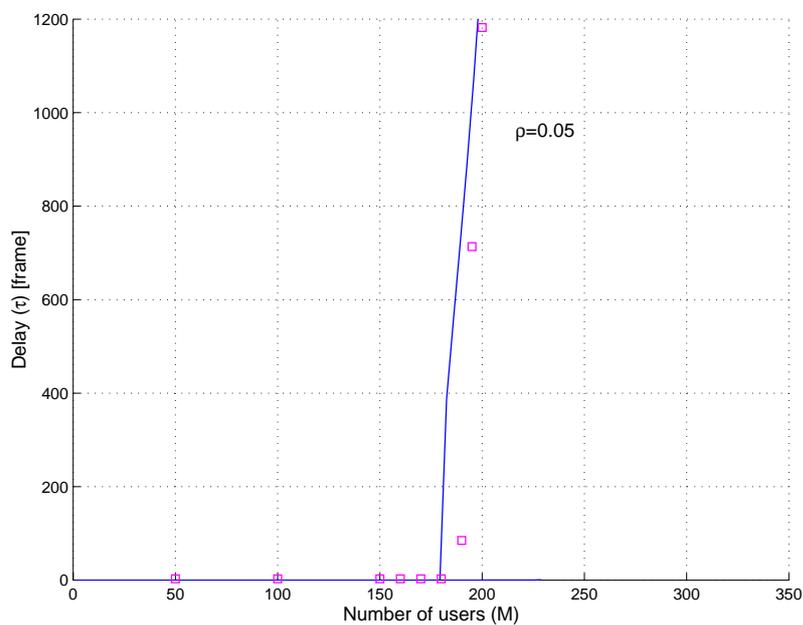
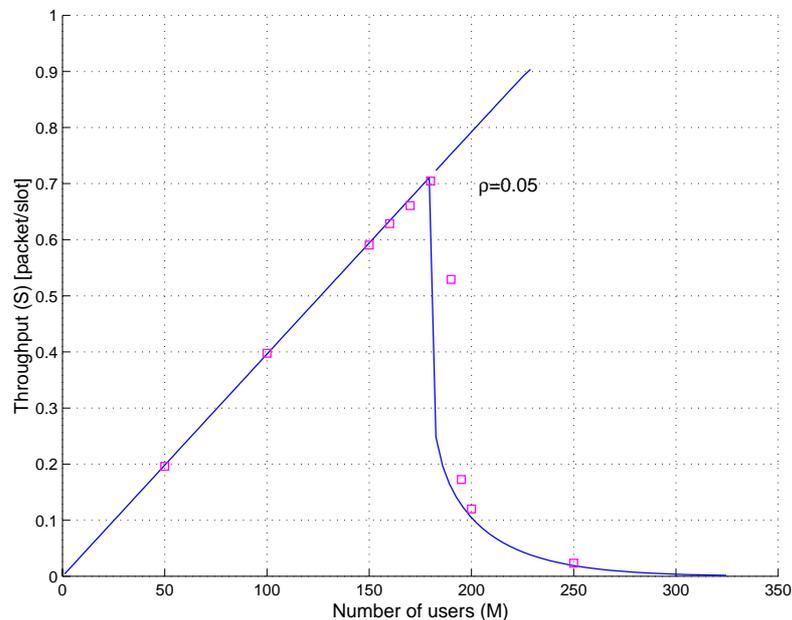
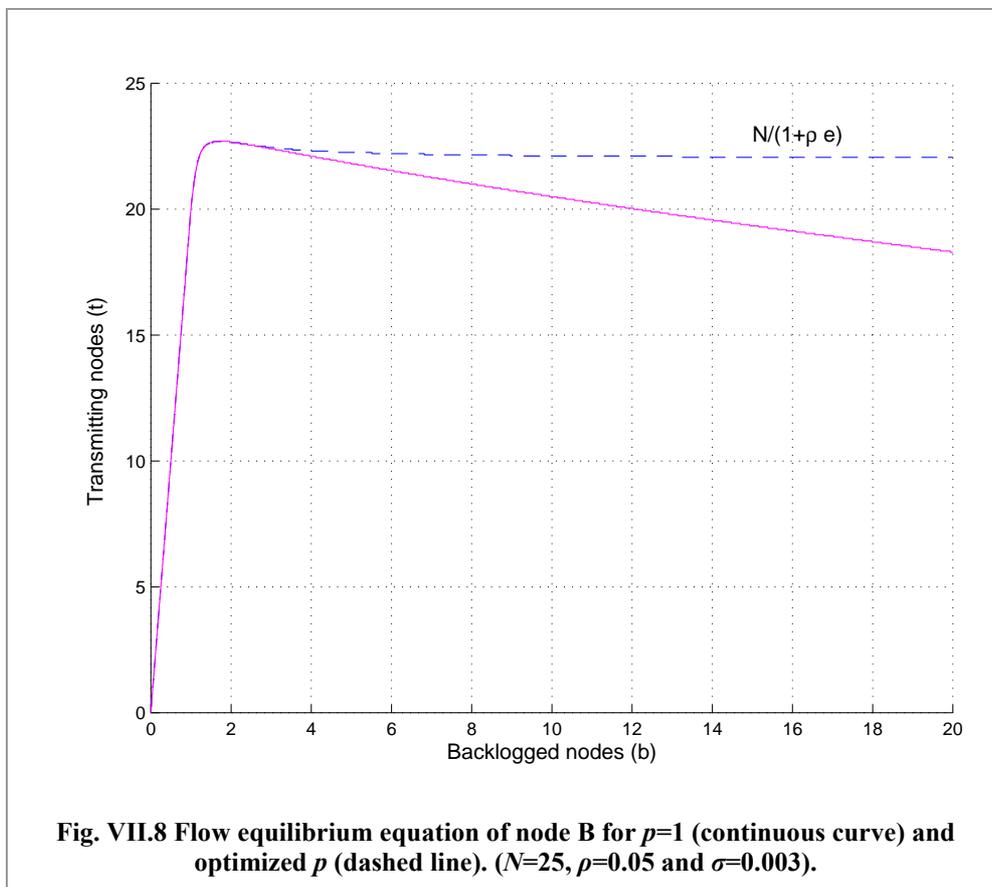


Fig. VII.6 C-TDMA performance with  $g=0.4$ ,  $N=25$  and permission probability  $p=0.2$ .



**Fig. VII.7 C-TDMA performance with  $g=0.0991$ ,  $N=25$  and permission probability  $p=0.9$ . Continuous line represents EPA results while square marks represent simulation results.**

Similar results are obtained for different values of the single user offered traffic  $g$ . The above curves remain roughly equal: the main differences is given by the number  $M$  of users giving the maximum throughput. Only for very low values of  $g$  the system shows bistability phenomena. An example is shown in Fig. VII.7, where in an intermediate range of  $M$  the throughput exhibits two values corresponding to a bistability situation. In this case the simulation agrees with the theory out of the bistability range, where on the contrary the simulated results assume intermediate values, corresponding to the fact that the system alternates periods of operation around the two stability points.



## VII.6. Optimization of Permission Probability

The permission probability  $p$  is a design parameter. In this section we discuss its optimal choice in terms of the other system parameters, namely the number  $M$  of users, the number  $N$  of slots per frame, and the traffic characteristics given by  $\rho$  and  $\sigma$ . Since  $p$  appears explicitly only into the output flow of the backlog state B (see (VII.16)), we

could proceed in the following recursive way. With an arbitrary value of  $p$  we find the equilibrium pair  $(b, t)$  of the backlogged and transmitting users; with these values the output flow of B depends on  $p$  and has a maximum for  $p' = (N - t)/b$ . Provided that the new value  $p'$  is meaningful ( $p \leq 1$ ), it is used as the permission probability and the new values of the equilibrium point are computed. The approach is iterated until convergence is reached.

An equivalent more direct approach consists in putting the value  $p = (N - t)/b$  into (VII.17) and resolve it so obtaining

$$t = \frac{N}{1 + \rho(1 - 1/b)^{1-b}} \quad (\text{VII.25})$$

This equation, combined with (VII.19), gives the solution  $(b, t)$ . If  $(N - t)/b < 1$ , this ratio is assumed as the optimal permission probability  $p_o$ . Otherwise, we put  $p_o = 1$ . Since  $b$  and  $t$  depend on the system parameters  $N$ ,  $\rho$ ,  $\sigma$  and  $M$ , also  $p_o$  is computable from these design data.

The advantage of using a permission probability is shown in Fig. VII.8. The flow equilibrium equation of node B is represented as a continuous curve for  $p = 1$  and as a dashed line for optimized  $p$ . Note that the curves coincide in the region  $N - t > b$ , where  $p_o = 1$ . It is worthwhile to note the presence of a horizontal asymptotical value of the throughput equal to  $N/(1 + \rho e)$  (see (VII.25)) for  $b \rightarrow \infty$ .

Usually the maximum throughput is slightly greater than the above asymptotical value; so we may adopt as a rule of thumb the formula

$$S_{\max} = \frac{N}{1 + \rho e},$$

independent of  $\sigma$ .

## VII.7. C–TDMA vs. PRMA: performance comparison

In this section we carry out a performance comparison between the C–TDMA and Packet Reservation Multiple Access (PRMA) [59], [58].

Even though PRMA applies with good performances to the speech scenario [59], also data transmission applications may be considered [58]. Of course, in order to compare on an analytical basis the proposed C–TDMA protocol with PRMA, we have to consider a simplified model for the last protocol in which the user cannot produce new packets until the previous packet is completed. This model can be applied to particular data transmission scenarios, in which the traffic emitted by the users is strictly dependent on the system state.

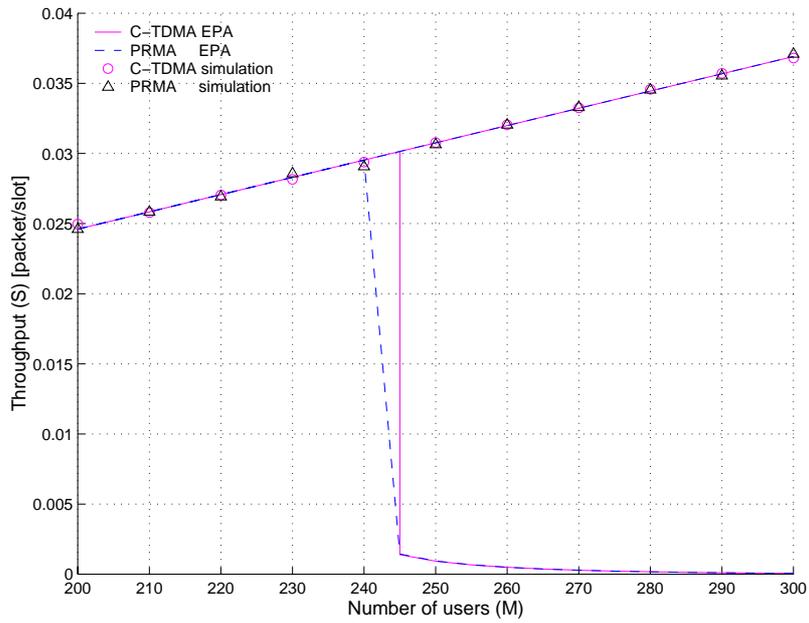
We have made some analytical comparisons between C–TDMA and PRMA in a data transmission scenario. Provided that the permission probabilities are chosen with sense for both protocols, they exhibit very similar performances. An example of comparison refers to parameters of Tab. VII.1, which we use to describe an interactive video application, in which intervals dedicated to the image vision alternate with requirements of a new video sequence. The permission probabilities are 1 for C–TDMA and 0.05 for PRMA. Fig. VII.9 shows the throughput evaluated by the EPA approach. With this choice of permission probabilities, both protocols exhibit bistability with increasing user number. The marked points denotes simulation results, which in the bistability region approximate the upper branch. This denotes that the probability of having a great number of users in contention state on the same time is very low with the parameters considered. Fig. VII.10 shows the average delay of the protocols. As it may be guessed by the previous comparison, also the delay performances are very similar.

**TAB. VII.1 PARAMETER VALUES FOR DATA TRANSMISSION SCENARIO (INTERACTIVE VIDEO)**

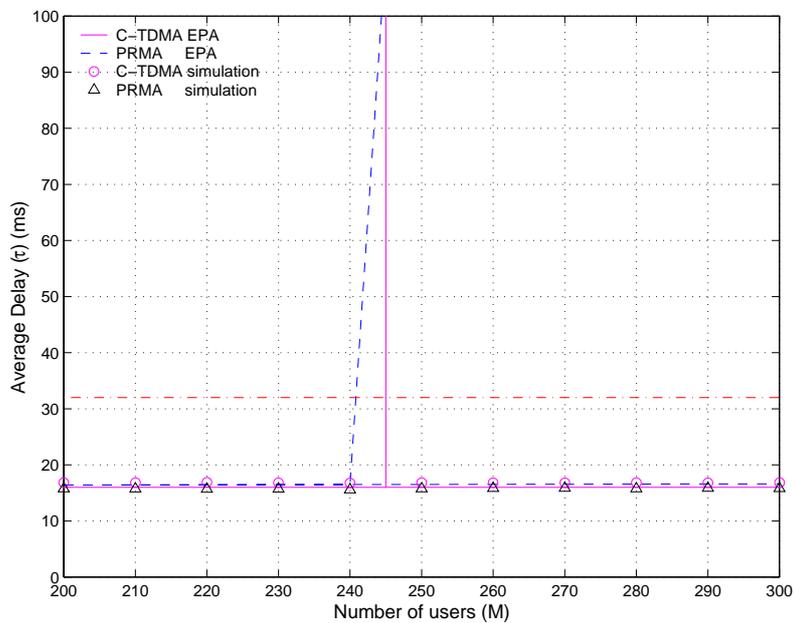
Parameter	Value
Mean message duration	0.288 s
Mean silence duration	120 s
Channel rate	720 kbit/s
Source rate	32 kbit/s
Frame duration	16 ms
Slots per frame	20
Slot duration	0.8 ms
Packet size	576 bits
Delay Constraint ( $D_{\max}$ )	unlimited

**TAB. VII.2 PARAMETER VALUES FOR SPEECH TRANSMISSION SCENARIO [59]**

Parameter	Value
Mean talkspurt duration	1.00 s
Mean silence duration	1.35 s
Channel rate	720 kbit/s
Source rate	32 kbit/s
Frame duration	16 ms
Slots per frame	20
Slot duration	0.8 ms
Packet size	576 bits
Delay Constraint ( $D_{\max}$ )	32 ms



**Fig. VII.9** Throughput of C-TDMA and PRMA as a function of the number of users  $M$  in the cell by using EPA and computer simulations (parameter values of Tab. VII.1).



**Fig. VII.10** Delay of C-TDMA and PRMA as a function of the number of users  $M$  in the cell by using EPA and computer simulations (parameter values of Tab. VII.1).

Clearly, the model so far considered cannot be used for applications (e.g. speech) in which the traffic emitted by the user is independent of the system state. For this case a more sophisticated PRMA model is required, as proposed by Nanda [59]. In the first model the system reacts to overloads by deferring the generation of new messages, while in the second model overloads cause packet dropping. On the other hand also C-TDMA could be modified in order to conveniently operate in a speech scenario. To this end, the system could allow users to produce new packets independently of the system state. The packets are stored in a buffer and dropped after a predefined contention delay. So far we do not have developed a mathematical analysis of this modified C-TDMA version. Then, we present only some comparisons between C-TDMA and PRMA performances in a speech scenario obtained by computer simulations.

Accordingly to the above discussion, we have compared the performance of the two protocols, by simulation, in a speech scenario. We have assumed the PRMA model considered by Nanda [59] with packet dropping after a maximum delay  $D_{max}$ . The C-TDMA model has been adapted to speech transmission by considering a traffic model that describes speech statistics, independently of the channel conditions. Furthermore, the model includes the packet dropping mechanism to discard packets that wait for a time longer than  $D_{max}$ . For both models the parameters, summarized in Tab. VII.2, coincide with the ones considered in [59]. The permission probability is  $p=0.3$ , as suggested in [59], for PRMA and  $p=1$  for C-TDMA.

Fig. VII.11 compares the mean and the standard deviation of the contention time vs. the number of users  $M$ . Owing to the fact that C-TDMA operates on a frame by frame basis, its average contention time is higher than for PRMA at least for low load, while for high load the performances tend to be similar. Also comparable are the standard deviations, depicted in the same figure with filled markers. Fig. VII.12 shows the packet dropping probabilities, the major measure of the system efficiency, vs. the number of users. The simulation results show that the two models exhibit substantially equivalent performance.

In conclusion, the fact that in the C-TDMA the base station send its information about the free slots only once per frame does not imply any significant penalty on the system performance.

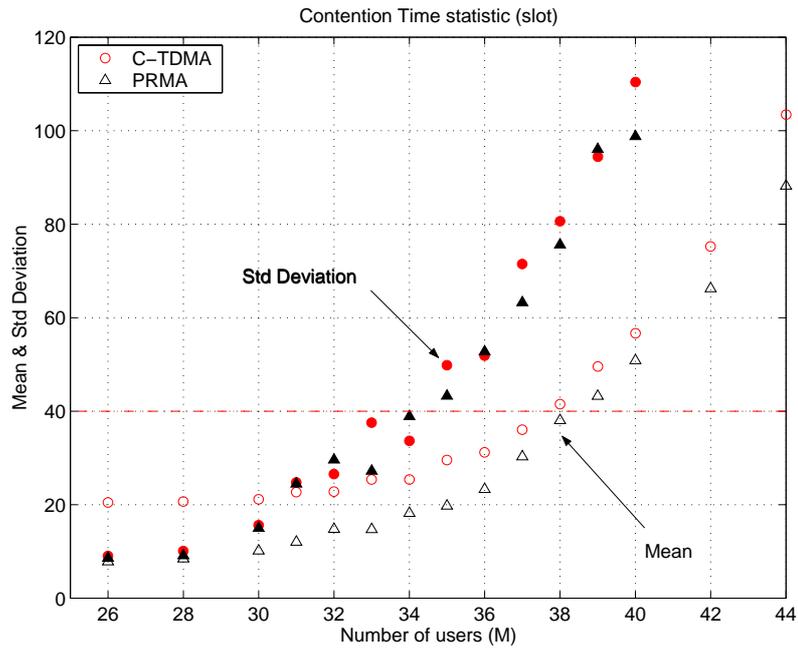


Fig. VII.11 Mean and standard deviation of the Contention Time for C-TDMA and PRMA vs. the number of users  $M$ , estimated by computer simulation in a speech scenario (Tab. VII.2).

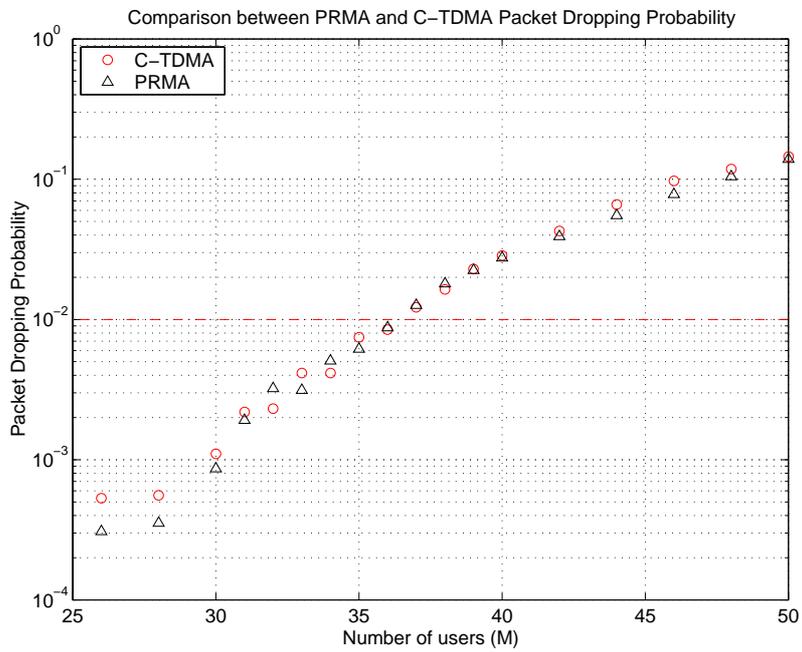


Fig. VII.12 Packet Dropping Probability for C-TDMA and PRMA vs. The number of users  $M$ , estimated by computer simulation in a speech scenario (parameter values of Tab. VII.2).

## VII.8. Conclusions

An access protocol for the up-link channel of cellular multi-user systems has been described. The protocol, named C-TDMA (Contention-Time Division Multiple Access), combines some properties of both the contention- and the reservation-based protocols. C-TDMA has been studied by using both classical Markov analysis and a simplified tool known as equilibrium point analysis (EPA), which for practical conditions (large number of users and channels) is more convenient, due to its reduced computational cost.

Performance evaluation of C-TDMA has been made in terms of throughput and delay by using EPA and computer simulations. These results demonstrate the accuracy of the EPA method and indicate that C-TDMA yields high throughput values with a limited delay in typical cellular scenarios. It has also been proposed a method of optimization of C-TDMA design to improve the maximum achievable throughput.

Performance comparison with PRMA in a data transmission scenario without real time constraints has shown that the protocols have very similar performances in terms of both throughput and delay. Finally, a simulation comparison with PRMA in a speech transmission scenario, with packet dropping, has shown that in this case C-TDMA achieves substantially equivalent performance to PRMA.

---

## APPENDIX VII-A

---

### AN OCCUPANCY PROBLEM

The determination of the statistics of  $z_n$  can be reduced to a classical occupancy problem [64]. Assume that distinguishable balls are put into  $n$  distinguishable cells, in such a way that, for each  $i = 1, \dots, n$ , the  $i$ -th cell can contain only either  $r_{i1}$  or  $r_{i2}$  or ... balls. Then, the series (possibly reducing to a polynomial)

$$p_i(\lambda) = \sum_j \frac{\lambda^{r_{i,j}}}{r_{i,j}!} \quad (\text{VII-A.1})$$

is named the enumerating function of the  $i$ -th cell. If the number of balls is  $m$ , the number of the different distributions of the balls into the cell according to the above constraints, is the coefficient of  $\lambda^m/m!$  in the following enumerating function [64]

$$E(\lambda) = \prod_{i=1}^n p_i(\lambda). \quad (\text{VII-A.2})$$

We are interested to the case in which  $K$  predetermined cells are occupied by one ball and the remaining  $n-k$  ones are either empty or occupied by more than one ball. This corresponds to successful transmissions in  $k$  predetermined slots. Then, the above constraints lead to the enumerating function

$$E(\lambda) = \lambda^k \left( 1 + \frac{\lambda^2}{2!} + \frac{\lambda^3}{3!} + \dots \right)^{n-k} = \lambda^k (e^\lambda - \lambda)^{n-k}. \quad (\text{VII-A.3})$$

After a simple algebra we find

$$E(\lambda) = \sum_{h=0}^{\infty} \sum_{i=k}^n \binom{n-k}{n-i} \binom{i+h}{h} i! (-1)^{i-k} (n-i)^h \frac{\lambda^{i+h}}{(i+h)!} \quad (\text{VII-A.4})$$

in which the coefficient of  $\lambda^m/m!$ , i.e. the number of dispositions of  $m$  balls leading to exactly  $k$  cells occupied by a single ball, is

$$N' = \sum_{i=k}^{\min\{n,m\}} \binom{n-k}{n-i} \binom{m}{i} i! (-1)^{i-k} (n-i)^{m-i}. \quad (\text{VII-A.5})$$

Hence, the number of dispositions leading to exactly  $k$  successes, independently of the cells wherein they happen, is

$$N = \binom{n}{k} N' = \sum_{i=k}^{\min\{n,m\}} \binom{n}{i} \binom{i}{k} \binom{m}{i} i! (-1)^{i-k} (n-i)^{m-i}. \quad (\text{VII-A.6})$$

Taking into account that the possible dispositions are  $n^m$ , the probability of exactly  $k$  successes becomes

$$p_k(n, m) = \frac{N}{n^m} = \sum_{i=k}^{\min\{n,m\}} \binom{n}{i} \binom{i}{k} \binom{m}{i} i! (-1)^{i-k} \frac{(n-i)^{m-i}}{n^m} \quad (\text{VII-A.7})$$

By applying the above result we get the conditioned probability that  $z_n$ , the number of backlogged nodes winning the contention, assume the value  $z$ . Indeed, we get

$$\begin{aligned} & \text{P}[z_n = z | t_n = t, b_n = b] \\ &= \sum_{c=0}^b \text{P}[z_n = z | c_n = c, t_n = t, b_n = b] \text{P}[c_n = c | t_n = t, b_n = b] \\ &= \sum_{c=z}^b p_z(N-t, c) \binom{b}{c} p^c (1-p)^{b-c} \\ &= \sum_{c=z}^b \sum_{i=z}^{\min\{N-t, c\}} \binom{N-t}{i} \binom{c}{z} \binom{c}{i} i! (-1)^{i-z} \frac{(N-t-i)^{c-i}}{(N-t)^c} \binom{b}{c} p^c (1-p)^{b-c} \quad (\text{VII-A.8}) \\ &= \sum_{i=z}^{\min\{N-t, b\}} \binom{N-t}{i} \binom{i}{z} i! (-1)^{i-z} \sum_{c=i}^b \binom{c}{i} \frac{(N-t-i)^{c-i}}{(N-t)^c} \binom{b}{c} p^c (1-p)^{b-c} \\ &= \sum_{i=z}^{\min\{N-t, b\}} \binom{N-t}{i} \binom{i}{z} \binom{b}{i} i! (-1)^{i-z} \frac{(N-t-pi)^{b-i}}{(N-t)^b} p^i. \end{aligned}$$

---

## VIII. TCP OVER WIRELESS LINKS

---

*Internet is based on the TCP/IP (Transmission Control Protocol/Internet Protocol) architecture. Unfortunately, the transmission of TCP/IP protocols over wireless networks may experience very poor performance. As a matter of fact, TCP Reno (TCPR), the more common TCP version at this time, cannot discriminate between packets dropped due to congestion and those dropped due to checksum test failure at the receiving side. Therefore, TCP invokes congestion recovery mechanisms even in absence of congestion, misinterpreting packet losses due to radio link unreliability. This chapter summarizes some approaches that have been proposed to improve the performance of TCP over wireless links. Furthermore, a new version of the TCP protocol, named TCP Westwood (TCPW) is presented and analyzed. Finally, an analytical model for TCPW connection is proposed and validated through computer simulations. The model is used to compare TCPW and TCPR over a single-bottleneck connection with different values of bottleneck capacity, round trip time, buffer size, and packet error probability.*

### VIII.1. Introduction: TCP congestion recovery mechanisms

The Transmission Control Protocol (TCP) provides end-to-end, reliable, congestion controlled connections over the Internet [17], [40]. Modern implementations of TCP contain four intertwined algorithms for congestion control and error recovery: slow start, congestion avoidance, fast retransmit, and fast recovery. A complete description of these mechanisms is given in [74][40][41][75], whereas [72] provides examples of the algorithms in action, and [73] provides the source code for the 4.BSD implementation.

Generally, the error recovery algorithms have been tuned to work properly in wired networks, where segment losses are mainly due to congestion. In the last period, however, has seen a progressive diffusion of wireless networking technologies. As a consequence, active research is in progress to extend the domain of effective TCP operability in such a new environment [6][9][10][11][45][33].

In order to understand the impact of wireless link characteristics on the end-to-end TCP performance, a deep knowledge of the congestion recovery mechanisms provided by TCP may be worthwhile. This section documents these algorithms for the Internet.

### **VIII.1.A. Slow Start and Congestion Avoidance**

In presence of congestion, TCP must slow down its transmission rate of packets into the network, and then invoke slow start to get things going again. Therefore, congestion avoidance and slow start algorithms are implemented together, even though they are logically independent entities with different objectives.

The assumption of algorithms is that packet loss caused by damage is very small (much less than 1%); therefore, the loss of a packet signals congestion somewhere in the network between the source and destination. Obviously, this assumption is not satisfied in radio environments and then it could cause many problems.

To implement these algorithms, two variables are added to the TCP per-connection state. The congestion window (cwnd) is a sender-side limit on the amount of data the sender can transmit into the network before receiving an acknowledgement (ACK); while the receiver's advertised window (rwnd) is a receiver-side limit on the amount of outstanding data. The minimum of cwnd and rwnd governs data transmission. The congestion window is flow control imposed by the sender, while the advertised window is flow control imposed by the receiver. The former is based on the sender's assessment of perceived network congestion; the latter is related to the amount of available buffer space at the receiver for this connection.

Another state variable, the slow start threshold (ssthresh), is used to determine whether the slow start or congestion avoidance algorithm is used to control data transmission, as discussed below.

Beginning transmission into a network with unknown conditions requires TCP to slowly probe the network to determine the available capacity, in order to avoid congesting the network with an inappropriately large burst of data. The slow start algorithm is used for this purpose at the beginning of a transfer, or after repairing loss detected by the retransmission timer.

The initial value of `ssthresh` may be arbitrarily high (for example, [79] suggests 64K, whereas some implementations use the size of the advertised window), but it may be reduced in response to congestion. The slow start algorithm is used when `cwnd` is below than `ssthresh`, while the congestion avoidance algorithm is used when `cwnd` is above `ssthresh`. When `cwnd` and `ssthresh` are equal, the sender may use either slow start or congestion avoidance.

### ***Slow Start Phase***

The slow start algorithm performs the following steps.

1. The initial value of `cwnd` is set less than or equal to  $2 \cdot \text{SMSS}$  (Sender Maximum Segment Size) bytes.
2. The congestion window is increased by one segment (at most SMSS bytes) for each ACK received that acknowledges new data. This provides exponential growth, although it is not exactly exponential because the receiver may delay its ACKs, typically sending one ACK for every two segments that it receives, to avoid the “Silly Window Syndrome” (see [57]).
3. The slow start phase ends when `cwnd` exceeds `ssthresh` or when congestion is observed. In the last case, the slow start phase is restarted with a new `ssthresh` value (see below) whereas in the former the congestion avoidance phase is started.

### ***Congestion Avoidance Phase***

Slow start has `cwnd` begin at one segment, and be incremented by one segment every time an ACK is received. As mentioned earlier, this opens the window exponentially: send one segment, then two, then four, and so on.

During congestion avoidance, `cwnd` is incremented by 1 full-sized segment per round-trip time (RTT). This is a linear growth of `cwnd`, compared to slow start's exponential growth. The increase in `cwnd` should be at most one segment each round-trip time (regardless how many ACKs are received in that RTT), whereas slow start increments `cwnd` by the number of ACKs received in a round-trip time. One formula commonly used to update `cwnd` during congestion avoidance is given in equation (VIII.1):

$$\Delta\text{cwnd} = \frac{\text{SMSS}^2}{\text{cwnd}} \quad (\text{VIII.1})$$

This adjustment is executed on every incoming non-duplicate ACK. Equation (VIII.1) provides an acceptable approximation to the underlying principle of increasing cwnd by 1 full-sized segment per RTT. Note that there are different implementations of cwnd increasing algorithm, some maintaining cwnd in units of bytes, while others in units of full-sized segments.

Congestion avoidance phase continues until congestion is detected.

### ***Exponential Backoff and Round Trip Estimation***

For each connection TCP maintains a variable, RTT, that is the best current estimate of the round-trip time to the destination. When a segment is sent, a timer is started with value set to the current Retransmission Timeout (*RTO*). If the ACK gets back before the timer expires, TCP measures how long the acknowledgment took, say *M*, and the difference between the expected and observed value,

$$Err = RTT - M, \quad (\text{VIII.2})$$

is computed. A smoothed value of this is maintained in *D* by the formula,

$$D = D + h(|Err| - D), \quad (\text{VIII.3})$$

which gives an approximation of the standard deviation of the acknowledgment inter-arrival time. Then, RTT is updated according to the formula

$$RTT = RTT + gErr. \quad (\text{VIII.4})$$

Finally, the timeout interval is set to the following value (in seconds):

$$RTO = \max\{RTO_{\min}; RTT + 4D\}, \quad (\text{VIII.5})$$

where *RTO<sub>min</sub>* is a TCP parameter and indicates the minimum value for the retransmission timeout.

Typical values for coefficient  $g$  and  $h$  and parameter  $RTO_{min}$  are the following,

- $g=1/8$ ;
- $h=1/4$ ;
- $RTO_{min}=250$  milliseconds.

If the timer goes off before receiving the acknowledgment for that packet, the packet has to be retransmitted and the value of the timer is doubled (exponential backoff). After that, the value of the timer is doubled for each retransmission with an upper limit of 64 seconds.

Note that, when a segment times out and is retransmitted, at the reception of relative ACK there is an ambiguity on RTT estimation. Indeed it is unclear whether the ACK refers to the first transmission or a later one. To avoid this inconvenient, the RTT is not updated on any segments that have been retransmitted, in accordance with the Karn's algorithm.

### ***Reaction against segment loss***

When a TCP sender detects segment loss using the retransmission timer, the value of  $ssthresh$  is set to no more than the value given in equation (VIII.6):

$$ssthresh = \max \{ \text{FlightSize}/2; 2 \cdot \text{SMSS} \}, \quad (\text{VIII.6})$$

where  $\text{FlightSize}$  denotes the amount of data that has been sent but not yet acknowledged. Except for particular cases, the  $\text{FlightSize}$  value coincides with  $cwnd$ .

Furthermore, upon a timeout  $cwnd$  is set to 1 full-sized. Finally the timer is adjusted second the exponential backoff strategy. Therefore, after retransmitting the dropped segment the TCP sender uses the slow start algorithm to increase the window from 1 full-sized segment to the new value of  $ssthresh$ , at which point congestion avoidance again takes over.

### **VIII.1.B. Fast Retransmit and Fast Recovery**

TCP may generate an immediate acknowledgement (a duplicate ACK) when an out-of-order segment is received. The purpose of this duplicate ACK (DACK) is to let the

other end know that a segment was received out of order, and to tell it what sequence number is expected. Such a feature is exploited by two algorithms, namely the *fast retransmission* and *fast recovery* algorithms ([41]), that were proposed in 1990 for improving the congestion recovery phase.

From the sender perspective, DACKs can be caused by a number of network problems: dropped segments, re-ordering of data segments by the network, replication of ACK or data segments by the network. Since TCP does not know what causes a DACK, it waits for a small number of DACKs to be received. It is assumed that if there is just a reordering of the segments, there will be only one or two DACKs before the reordered segment is processed, which will then generate a new ACK. If three or more duplicate ACKs are received in a row (4 identical ACKs without the arrival of any other intervening packets), it is a strong indication that a segment has been lost. TCP then performs a retransmission of what appears to be the missing segment, without waiting for a retransmission timer to expire (fast retransmission).

After the fast retransmit algorithm sends what appears to be the missing segment, the fast recovery phase is started. In this phase, the sender side keeps transmitting segments as in the congestion avoidance phase, i.e. one for each DACK received. The receipt of the DACKs, indeed, indicates that there is still data flowing between the two ends, and TCP does not want to reduce the flow abruptly by going into slow start. Furthermore, since the ACK “clock” [40] is preserved, the TCP sender can continue to transmit new segments (although transmission must continue using a reduced cwnd).

The fast retransmit and fast recovery algorithms are usually implemented together as follows.

1. When the third duplicate ACK is received, set *ssthresh* to no more than the value given in equation (VIII.6).
2. Retransmit the lost segment and set *cwnd* to *ssthresh* plus  $3 \cdot \text{SMSS}$ . This artificially “inflates” the congestion window by the number of segments (three) that have left the network and which the receiver has buffered.
3. For each additional duplicate ACK received, increment *cwnd* by *SMSS*. This artificially inflates the congestion window in order to reflect the additional segment that has left the network.

4. Transmit a segment, if allowed by the new value of `cwnd` and the receiver's advertised window.
5. When the next ACK arrives that acknowledges new data, set `cwnd` to `ssthresh` (the value set in step 1). This is termed "deflating" the window. This ACK should be the acknowledgement elicited by the retransmission from step 1, one RTT after the retransmission. Additionally, this ACK should acknowledge all the intermediate segments sent between the lost segment and the receipt of the third duplicate ACK, if none of these were lost.

## VIII.2. Improving TCP performance over wireless links

In this section, we outline some approaches that have been proposed to improve the performance of TCP over wireless links.

There are three fundamentally different approaches to improving TCP performance in lossy systems, which can be classified in the following families:

- end-to-end;
- split connection;
- link layer.

Techniques of the first kind attempt to make the sender aware of the existence of wireless links, in order to avoid invoking congestion control algorithms when non-congestion-related losses occur.

The second approach consists in splitting the end-to-end connection into two parts at the interface between the wired and wireless trunks: the classic TCP protocols may be used for the wired section, while a special-purpose transmission protocol may cover the wireless section.

Finally, the third solution acts at link layer and requires no change in the existing TCP implementations. The intuition behind this approach is that, since the problem is local, it should be solved locally: the transport layer need not to be aware of the characteristics of the individual links. Link layer protocols, thus, attempt to shield the upper layer applications from the unreliability of the lossy link, offering a higher quality data link connection with a reduced effective bandwidth.

Following subsections present a rapid overview of these strategies, with some example of protocols of different classes.

### **VIII.2.A. End-to-end approach**

Although a wide variety of TCP versions are used on the Internet, the current de facto standard for TCP implementations is TCP Reno [72]. A possible method to improve the performance of TCP Reno after multiple packet losses in a window, consists in adding selective acknowledgments (SACKs) to the standard TCP Reno stack. This allows the sender to handle multiple losses within a window of outstanding data more efficiently. However, the sender still assumes that losses are a result of congestion and invokes congestion control procedures, such as shrinking its congestion window size.

Another proposed strategy consists in adding Explicit Loss Notification (ELN) option to TCP acknowledgments [8]. When a packet is dropped on the wireless link, future cumulative acknowledgments corresponding to the lost packet are marked to identify that a non-congestion related loss has occurred. Upon receiving this information with duplicate acknowledgments, the sender may perform retransmissions without invoking the associated congestion-control procedures.

The main drawback of end-to-end solutions is that, generally, they require to modify TCP protocol at both sender and receiver side. Of course, considering the huge number of computers connected to the Internet, modifying the TCP/IP stack on a global base requires an important effort.

### **VIII.2.B. Split connection approach**

Split-connection approaches completely hide the wireless link from the sender by terminating the TCP connection at the base station. Such schemes use a separate reliable connection between the base station and the destination host. The second connection can use techniques such as negative or selective acknowledgments, rather than just regular TCP, to perform well over the wireless link.

One of the early protocols to use the split-connection approach is the *Indirect-TCP* (I-TCP) protocol [7]. It involves splitting each TCP connection between a sender and

receiver into two separate connections at the base station: one TCP connection between the sender and the base station, and the other between the base station and the receiver.

I-TCP, like other split-connection proposals, attempts to separate loss recovery over the wireless link from that across the wired line network, thereby shielding the original TCP sender from the wireless link. However, as some experiments indicate, the choice of TCP over the wireless link results in several performance problems. Since TCP is not well tuned for the lossy link, the TCP sender of the wireless connection often times out, causing the original sender to stall. In addition, every packet incurs the overhead of going through TCP protocol processing twice at the base station (as compared to zero times for a non-split-connection approach), although extra copies are avoided by an efficient kernel implementation.

Another disadvantage of this approach is that the end-to-end semantics of TCP acknowledgments is violated, since acknowledgments to packets can now reach the source even before the packets actually reach the mobile host. Also, since this protocol maintains a significant amount of state at the base station per TCP connection, handoff procedures tend to be complicated and slow.

### **VIII.2.C. Link-layer approach**

The third class of protocols, link-layer solutions, lies between the other two classes. These protocols attempt to hide link-related losses from the TCP sender by using local retransmissions and perhaps forward error correction over the wireless link. The local retransmissions use techniques that are tuned to the characteristics of the wireless link to provide a significant increase in performance.

Since the end-to-end TCP connection passes through the lossy link, the TCP sender may not be fully shielded from wireless losses. This can happen either because of timer interactions between the two layers [22], or TCP's duplicate acknowledgments, which cause sender fast retransmissions even for segments that are locally retransmitted. As a result, some proposals to improve TCP performance use mechanisms based on the knowledge of TCP messaging to shield the TCP sender more effectively and avoid competing and redundant retransmissions [11]. An example of this kind of protocol is given by the *Snoop Protocol* [39].

The snoop protocol introduces a module, called the *snoop agent*, at the base station. The agent monitors every packet that passes through the TCP connection in both directions and maintains a cache of TCP segments sent across the link that have not yet been acknowledged by the receiver. A packet loss is detected by the arrival of a small number of duplicate acknowledgments from the receiver or by a local timeout. The snoop agent retransmits the lost packet if it has it cached and suppresses the duplicate acknowledgments.

The main advantage of this approach is that it suppresses duplicate acknowledgments for TCP segments lost and retransmitted locally, thereby avoiding unnecessary fast retransmissions and congestion control invocations by the sender. Like other link-layer solutions, the snoop approach could also suffer from not being able to completely shield the sender from wireless losses.

Generation of duplicate acknowledgments is due to utilization of ARQ techniques on TCP segments. Other link-layer protocols could avoid this drawback implementing the ARQ mechanism in an ad-hoc software layer underneath the TCP/IP layer, in such a way that retransmissions involve low level data structures (e.g., radio packet). An example of such a solution is given by the ARQ mechanism adopted by the Bluetooth standard (see Section III.5).

This kind of solutions presents the great advantage of not requiring any knowledge of the above data structures; on the other hand, the retransmission mechanism could strongly modified the link delay, producing spurious retransmission at TCP level due to timers expiration.

### **VIII.3. TCP Westwood: an end-to-end approach**

TCP Westwood [16] design adheres to the end-to-end transparency guidelines set forth in [38]. Namely, a simple modification of the TCP source protocol stack which allows the source to estimate the available bandwidth, and to use the bandwidth estimation to recover faster, thus achieving higher throughput. TCP Westwood exploits two basic concepts: the end-to-end estimation of the available bandwidth, and the use of such estimate to set the slow start threshold and the congestion window. It is worth

underscoring that TCPW does not require any intervention from network layer or proxy agents.

TCPW source continuously estimates the packet rate of the connection by properly averaging the rate of returning ACKs. The estimate is used to compute the congestion window and slow start threshold to be used after a congestion episode is detected, that is, after three duplicate acknowledgments or after a timeout. The rationale of this strategy is simple: in contrast with TCP Reno, which simply halves the congestion window after three DACKs, TCP Westwood (TCPW) attempts to make a more “informed” decision. It selects a slow start threshold and a congestion window that are consistent with the effective connection rate at the time congestion is experienced. We call such mechanism *very faster recovery*.

The use of bottleneck bandwidth and connection rate estimation has been proposed before in the TCP literature. The best known examples are Packet Pair (PP) [4] and TCP Vegas [49]. Both of these schemes use the bandwidth computation to estimate the bottleneck backlog. The larger the backlog, the larger the congestion. The backlog feedback is used in both cases to control the send window. The PP scheme achieves perfect bottleneck fair sharing. Unfortunately, it requires per flow queuing and Round Robin scheduling – a feature not available in commercial routers.

Also related to this work is the probing of “available bandwidth” on a path. Allman and Paxson in [17] report and compare techniques for probing the available bandwidth in order to properly initialize the *ssthresh* before a TCP connection is started. An excessively large *ssthresh* can lead to premature timeout, slow start and efficiency loss. Lai and Baker in [40] describe an improved measurement technique (packet tailgating) to probe available bandwidth that is less intrusive (i.e., consumes less bandwidth) than previous techniques. The above techniques share with our scheme the notion of measuring the interpacket delay gaps and deriving bandwidth information from it. They do, however, probe the available bandwidth *before* the connection is started. This is a very different (and potentially much more difficult) problem than measuring the actual rate that a connection is achieving *during the data transfer*.

The “key innovation” of TCPW is to use the bandwidth estimate “directly” to drive the window, instead of using it to compute the backlog. The rationale is that if a

connection is currently achieving a given rate, then it can safely use the window corresponding to that rate without causing congestion in the network.

In this chapter we focus on the behavior of TCPW in packet loss environments, caused by link error, wireless interference or buffer overflow. Like Reno, TCPW cannot distinguish between buffer overflow loss and random loss. However, in presence of random loss, TCP Reno overreacts and reduces the window by half. TCPW, on the other hand, after packet loss and retransmission timeout, resumes with the previous window as long as the bottleneck is not yet saturated (i.e., no buffer overflow).

It may be worthwhile, finally, underline that TCPW does not require any specific intervention/support from intermediate route, thus preserving the original “end-to-end design” principle.

In order to study the behavior of TCPW in presence of random errors we have derived an analytic model, obtained by using Markov techniques. The model has been inspired by previous works on TCP Reno. In this case, however, we have considered the bandwidth estimation mechanism provided by TCPW, an element that was not present in the previous works.

This model is an important contribution in that it provides further insight in TCPW operation. It allows us to investigate the performance achieved by the TCPW protocol in different scenarios and cross validate simulation and measurements.

### **VIII.3.A. Overview of TCP Westwood**

In this section we give a short overview of the operating of TCP Westwood. A detailed description of the algorithm can be found in [16].

First, we note that in TCPW congestion window increments during slow start and congestion avoidance remain the same as in Reno, that is they are exponential and linear respectively. The only difference between the two protocols is the manner in which they react to a packet loss.

In TCP Westwood the sender continuously computes the connection BandWidth Estimate (BWE) that is defined as the share of bottleneck bandwidth used by the connection. Thus, BWE is equal to the rate at which data is delivered to the TCP

receiver. The estimate is based on the rate at which ACKs are received and on their payload. We can measure the sample bandwidth used by that connection as  $b_k = d_k / \Delta t_k$ , where  $d_k$  is the amount of data notified by an ACK and  $\Delta t_k$  is the time elapsed since previous ACK was received. Since congestion occurs whenever the low-frequency input traffic rate exceeds the link capacity [52], a low-pass filter is employed to average sampled measurements and to obtain the low-frequency components of the available bandwidth.

After a packet loss indication, i.e. reception of 3 duplicate ACKs (DACKs), or timeout expiration, the sender resets the congestion window (cwnd) and the slow start threshold (ssthresh) as follows:

```
if (3 DACKs are received) then
{
    ssthresh = (BWE * RTTmin) / seg_size;
    if (cwnd > ssthresh) then /* congestion avoid. */
    {
        cwnd = ssthresh;
    }
}
```

In the case a packet loss is indicated by a timeout expiration, cwnd and ssthresh are set as follows:

```
if (coarse timeout expires) then
{
    cwnd = 1;
    ssthresh = (BWE * RTTmin) / seg_size;
    if (ssthresh < 2) then
    {
        ssthresh = 2;
    }
}
```

The rationale of the algorithm above is that after a timeout, cwnd and the ssthresh are set equal to 1 and BWE, respectively. Thus, the basic Reno behavior is still captured, while a speedy recovery is ensured by setting ssthresh to the value of BWE.

Besides the bandwidth estimation, an important element of this procedure is the RTT estimation. RTT is required to compute the window that supports the estimated rate

BWE. Ideally, the RTT should be measured when the bottleneck is empty. In practice, it is set equal to the overall minimum round trip delay (RTT<sub>min</sub>) measured so far on that connection (based on continuous monitoring of ACK RTTs).

### VIII.3.B. TCPW analytic model

In this section we develop an analytic model of TCPW congestion control mechanism. The objective is to study the throughput taking into account the impact of the filter operation, in addition to the following system parameters: bottleneck link transmission speed, round trip time, and link error rate.

#### *System Description*

Following [89], [50], and [3] we consider a saturated TCP source that always has fixed-size packets to send, of length  $L$ . The sender releases packets into a limited FIFO buffer that can hold up to  $B$  packets (packet arriving to a full buffer are discarded). The packets are then sent over a single bottleneck link with a speed of  $\mu$  packets per second and a two way propagation delay of  $d$ , where  $d$  is assumed deterministic and consists of propagation time and any other processing delays excluding the transmission time and the bottleneck link queuing delay. Let  $T$  denote the time between the beginning of a packet transmission and the reception of an ACK for the same packet, excluding the queuing delay in the buffer. Then  $T=d+1/\mu$ . Since a packet loss would occur once the buffer is full, the maximum window size a connection can achieve is  $W_{max}=B+\mu T=B+C$ , where  $C = \mu T$  is the *pipe capacity*.

The throughput realized by the system can be derived from the time evolution of the congestion window,  $W(t)$ . The evolution of this parameter has a cyclic structure, in which  $W(t)$  grows until a packet loss is detected. At this point, the missed packet is retransmitted and the transmission stops until the cumulative ACK for the last window of packets is received. The cycle, then, starts again with the congestion window and the slow start threshold set according to the bandwidth estimation at the instant in which the missed packet was detected.

In order to make the mathematical analysis feasible, a simplified model of the system is considered with the following assumptions:

1.  $d$  and  $\mu$  constants;
2. Independent packet errors with probability  $v$ ;
3. Single packet loss in each cycle;
4. System always in the congestion avoidance phase;
5. Error-free feedback path;
6. Exponential timer backoff, fast recovery ignored.

By assumption 3, retransmitted packets are never dropped and the buffer overflow at the bottleneck link produces a single packet loss. Furthermore, by assumption 4 we assume that the congestion window at the end of a cycle is always greater than or equal to the slow start threshold of the following cycle. With very high error probability, these assumptions may not be satisfied. However, the close match between theoretical and simulation results that will be shown in the following prove that these approximations are not crucial and do not affect the accuracy of the model.

In the next subsection, we describe the cycle evolution under the hypotheses considered. Then, we describe the evolution of the congestion window at the beginning of successive cycles through a Markovian model and we derive an expression for the average system throughput.

### ***Cycle evolution***

Let the initial window size of the  $m$ -th cycle be denoted by  $W_0^{(m)}$ ; also let  $N_{drop}^{(m)}$  be the index of the packet lost at the end of the cycle. The evolution of the cycle is then determined by these parameters. For the sake of clarity, the reference to the cycle number  $m$  will be omitted in the following, whenever unnecessary. During a cycle, acknowledgments arrive in a series of consecutive *bursts*. ACKs in the same burst arrive  $1/\mu$  seconds apart while consecutive bursts arrive  $T$  second apart, until the pipe capacity,  $C$ , is reached. After this point, ACKs arrive continuously every  $1/\mu$  seconds. The congestion window is increased by one at the end of each burst.

If  $W_0$  is the initial congestion window, the first burst contains exactly  $W_0$  packets, the second  $W_0+1$  and so on. Let  $B_k$  identify the  $k$ -th burst of a cycle. We denote the total

number of ACKs in the burst  $B_k$ , or the *burst length*, by  $l_k$ . The burst length is related to  $W_0$  and  $k$  as follows:

$$l_k = W_0 + k - 1. \quad (\text{VIII.7})$$

Let  $B_{k^*}$  be the burst at which the pipe capacity is reached.  $k^*$  can be determined from  $k^* = C - W_0 + 1$ . After we reached the pipe capacity, packets are buffered and ACKs arrive continuously,  $1/\mu$  second apart. At the end of the burst number  $k^{**} = k^* + B$ , the congestion window is set to  $W_{max} + 1$ , and the last packet sent is discarded due to buffer overflow. Skipping the details of the fast retransmission and recovery mechanisms, we assume that the cycle ends  $(C+B)/\mu$  seconds later, when the ACK for the dropped packet is missed.

We number the packets sent in a cycle taking into account the bursts  $B_k$ . Let  $s_k$  be the first packet of  $B_k$ .  $s_k$  can be determined as follows:

$$s_k = 1 + \sum_{j=1}^{k-1} l_j = 1 + (W_0 - 1)(k - 1) + \frac{k(k-1)}{2}. \quad (\text{VIII.8})$$

Let  $n_{of}$  be the packet dropped due to buffer overflow. It is related to  $W_0$  through the following formula:

$$n_{of}(W_0) = s_{k^*} + 2(C + B). \quad (\text{VIII.9})$$

Denoting with  $k_n$  the number of the burst that contains the packet number  $n$ , and with  $r_n$  the offset of the packet in the burst  $B_{k_n}$ , we can express each  $n$ ,  $1 \leq n \leq n_{of}$ , as  $n = s_{k_n} + r_n$ , with

$$k_n = \left\lfloor -W_0 + \frac{3}{2} + \sqrt{W_0^2 - W_0 - \frac{7}{4} + 2n} \right\rfloor; \quad (\text{VIII.10})$$

$$r_{k_n} = n - s_{k_n}.$$

For simplicity, we assume that the congestion window is set according to the bandwidth estimation at that instant and that a new window worth of packets is immediately sent.

The instant in which the  $n$ -th ACK is received is, then, given by

$$t(W_0, n) = \begin{cases} Tk_n + \frac{r_n}{\mu} & n \leq s_{k^*} \\ Tk^* + \frac{n - s_{k^*}}{\mu} & n > s_{k^*} \end{cases} \quad (\text{VIII.11})$$

### ***Slow start threshold evolution***

At the beginning of a generic cycle  $m$ , TCPW sets the slow start threshold (and the congestion window) according to the value of the estimated bandwidth at the end of the previous cycle:

$$W_0^{(m)} \stackrel{\text{def}}{=} f_W(W_0^{(m-1)}, N_{drop}^{(m-1)}) \cong \max \left\{ 2, \left\lfloor \frac{BWE(W_0^{(m-1)}, N_{drop}^{(m-1)})T}{L} \right\rfloor \right\} \quad (\text{VIII.12})$$

where  $N_{drop}^{(m-1)}$  is the ACK missed at the end of cycle  $(m-1)$ . The TCPW bandwidth can be modeled by:

$$BWE(n+1) \cong \alpha BWE(n) + (1-\alpha) \frac{b_{n+1} + b_n}{2}, \quad (\text{VIII.13})$$

where  $\alpha$  is related to the cut-off frequency of the low pass filter, while  $BWE(0)$  is set to the bandwidth estimated at the end of the previous cycle. In (VIII.13),  $b_n$  is given by  $b_n = d_n / \Delta t_n$ , where  $d_n$  is the amount of data acknowledged by the ACK for the  $n$ -th packet sent in that cycle, while  $\Delta t_n$  is the time elapsed since the reception of the previous ACK. On the basis of the previous observations about the ACK inter-arrival times, the bandwidth information carried by the  $n$ -th ACK is then given by:

$$b_n = \begin{cases} \frac{\mu}{C - l_{k_{n-1}} + 1} & n = s_{k_n}, k_n < k_C \\ \mu & \text{otherwise} \end{cases} \quad (\text{VIII.14})$$

Substituting (VIII.14) in (VIII.13) and developing the recursion we obtain the bandwidth estimated after each ACK arrival.

### Throughput estimation

On the basis of the formulas introduced in the previous subsections, it is easy to realize that the evolution of the initial congestion window size, in each cycle, can be modeled as a Markov process. Indeed,  $W_0^{(m)}$  depends only on  $W_0^{(m-1)}$  and  $N_{drop}^{(m-1)}$ , which is once again a function of the packet error probability and of  $W_0^{(m-1)}$ . Thus, the steady state probability vector for the initial window size can be derived from the state transition probability matrix of the process.

Let  $P_{i,j}$  be the transition probability from  $W_0^{(m-1)}=i$  to  $W_0^{(m)}=j$ :

$$P_{i,j} = \Pr\{W_0^{(m)} = j | W_0^{(m-1)} = i\}; \quad i, j \in \{2, 3, \dots, C\} \quad (\text{VIII.15})$$

The explicit computation of (VIII.15) would require the inversion of equation (VIII.13), in order to determine the value of  $N_{drop}^{(m-1)}$  for which the bandwidth estimation at the end of the cycle corresponds to a window size of  $W_0=j$ , starting with a window  $W_0=i$ . Unfortunately, the equation cannot be inverted. Hence, the transition matrix is computed row by row, in the following way:

$$P_{i,j} = \sum_{n \in I_j(i)} p_{i,n}; \quad i=2, \dots, C \quad (\text{VIII.16})$$

where  $p_{i,n}$  is the probability that the packet  $n$  is dropped, given that the initial window size is  $i$ , while  $I_j(i)$  is the set of values of  $n$  for which the estimated bandwidth at the step  $n$  corresponds to a window size of  $j$ .

Explicitly, we have

$$p_{i,n} = \Pr[N_{drop} = n | W_0 = i] = \begin{cases} v(1-v)^{n-1}, & n < n_{of}(i) \\ (1-v)^{n-1}, & n = n_{of}(i) \end{cases} \quad (\text{VIII.17})$$

and

$$I_j(W_0) = \{ n: 1 \leq n \leq n_{of}(W_0), f_w(n, W_0) = j \}. \quad (\text{VIII.18})$$

Finally, denoting with  $\pi_w$  the asymptotic probability of the initial window size  $W$ , the average throughput realized by the system is given by

$$\bar{\eta} = \sum_{W_0=2}^C \pi_{W_0} \sum_{n=1}^{n_{of}(W_0)} \frac{n-1}{t(W_0, n)} P_{W_0, n}. \quad (\text{VIII.19})$$

It is worth noticing that the same approach can be applied to derive an analytical model for TCP Reno. In this case, the transition probability  $P_{i,j}$  is given by

$$P_{i,j} = \sum_{n \in B_{2^j-1+i}} p_{i,n}; \quad i = 2, \dots, C. \quad (\text{VIII.20})$$

## VIII.4. Performance Evaluation

In this section, we use the model described above to study the performance of TCPW and compare it with that of TCP Reno. The analysis is carried out considering the general case of a lossy channel as well as the ideal case of an error-free link.

A concise summary of the system parameters selected in our analysis is provided in Tab. VIII.1

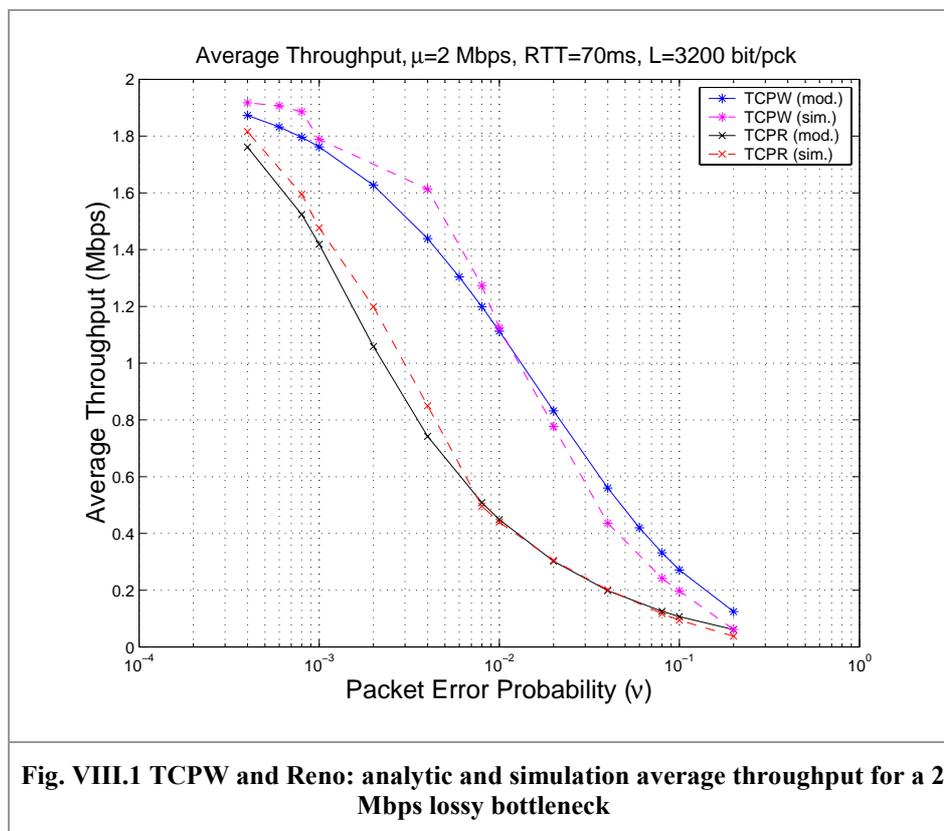
Parameters	Lossy Link (2Mbps)
Packet- length	400
Link capacity $\mu$ (pkts/sec)	625
Round Trip Delay (T)	70 msec
Buffer size (B)	44
$\alpha$	19/21

Link speed, packet length and round trip delay are typical, while the buffer size is set to the pipe capacity  $C = \mu T$ . The choice of the parameter  $\alpha$  requires particular care. Basically,  $\alpha$  represents the pole of the low-pass filter of equation (VIII.13), and it has to be large enough to properly smooth the bandwidth estimations. On the other hand, the response time of the filter must be short enough for the filter to quickly reach steady

state bandwidth estimation. If the initial bandwidth estimation is zero and the input bandwidth samples carry a value of  $\mu$ , the output of the filter is approximately:

$$BWE(n) = \mu(1 - \alpha^n)$$

We can determine an upper bound on  $\alpha$  that will ensure that the relative error in the bandwidth estimation after the reception of  $n$  samples is less than  $k$ . This condition leads to  $\alpha^C < k$  which, for our choice of  $k=0.1$ ,  $n=C$  and  $\alpha=19/21$ , is satisfied.



### Lossy connection

Fig. VIII.1 shows the average throughput attained by TCP Westwood and Reno, evaluated using simulations and through the application of the analytic model over a wireless 2Mbps bottleneck. The first point of interest is the close match between analytic and simulation results for both the TCP protocols. The second and perhaps more significant result is the higher throughput TCPW achieves throughout a wide range of packet error rate (PER) values. The two protocol throughputs converge in the

limiting cases when either the errors are very frequent (trivially, the throughput tends to zero) or very rare (this last case will be elaborated upon in subsection VIII.4.A).

#### VIII.4.A. Ideal error-free connection

In the previous subsection, we analyzed TCPW performance in the general case of a lossy link bottleneck. In order to study the limiting behavior of the protocol, in addition to its sensitivity to the main system parameters, we also investigate the TCPW performance under the ideal condition of an error-free channel. The results presented in this subsection are obtained taking advantage only of the analytic model of TCPW developed in the previous section.

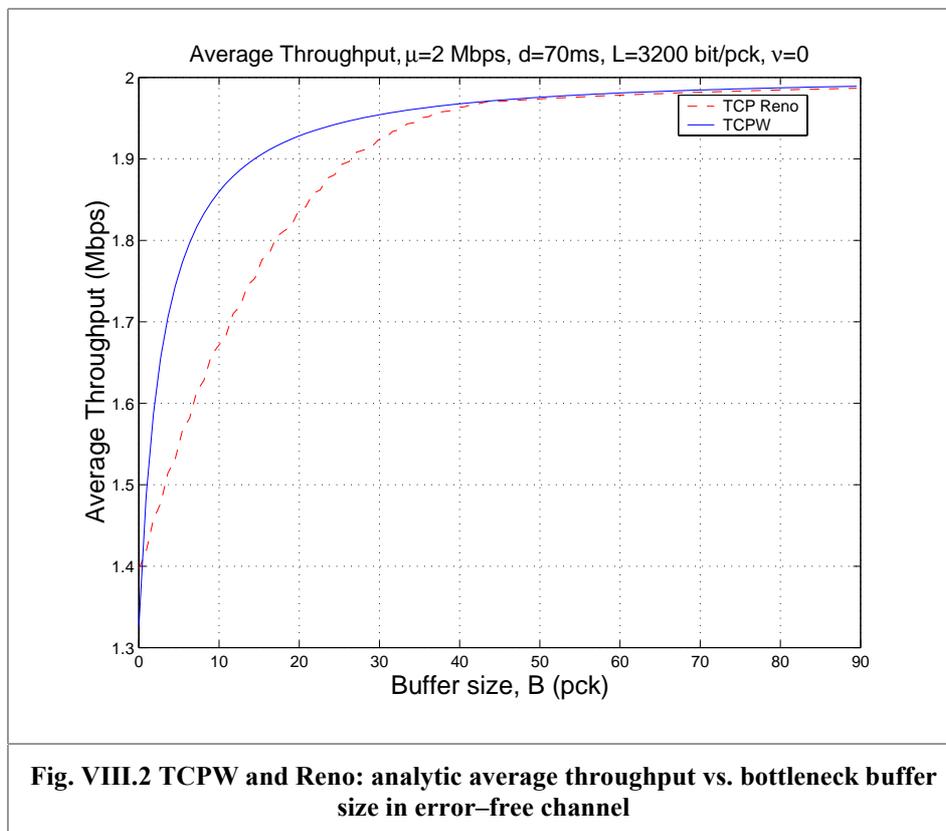


Fig. VIII.2 shows the average throughput against the bottleneck buffer size  $B$ . We note that the throughput achieved by TCP Reno is always below that of TCPW, while the two curves are almost superimposed for values of  $B$  greater than the pipe size. The rationale of this result is that TCPW, due to its bandwidth estimation algorithm, starts each cycle filling the pipe capacity; while, TCP Reno halves its window size. Hence, for  $B < C$ , the new initial window is not sufficient to saturate the link. On the other hand, for

$B > C$  the new window is greater than the pipe capacity and the performance is maximized.

Finally, it is worth emphasizing that increasing  $B$  beyond the value of  $C$  does not result in a considerable gain in terms of throughput and this justifies our choice of  $B = C$ .

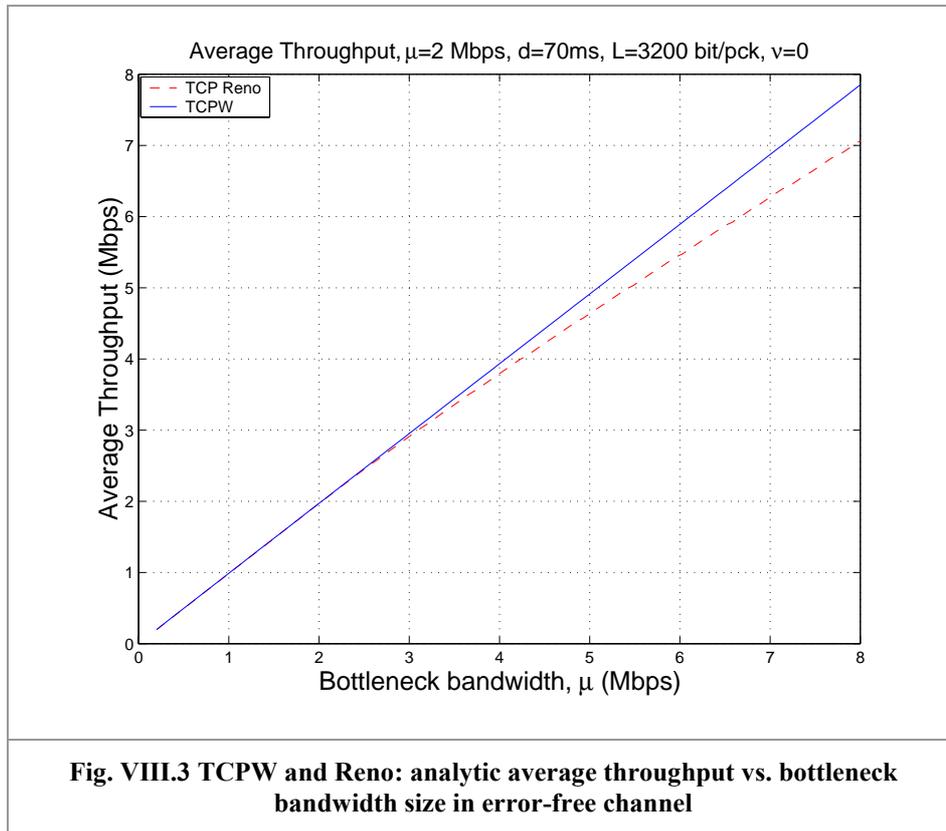
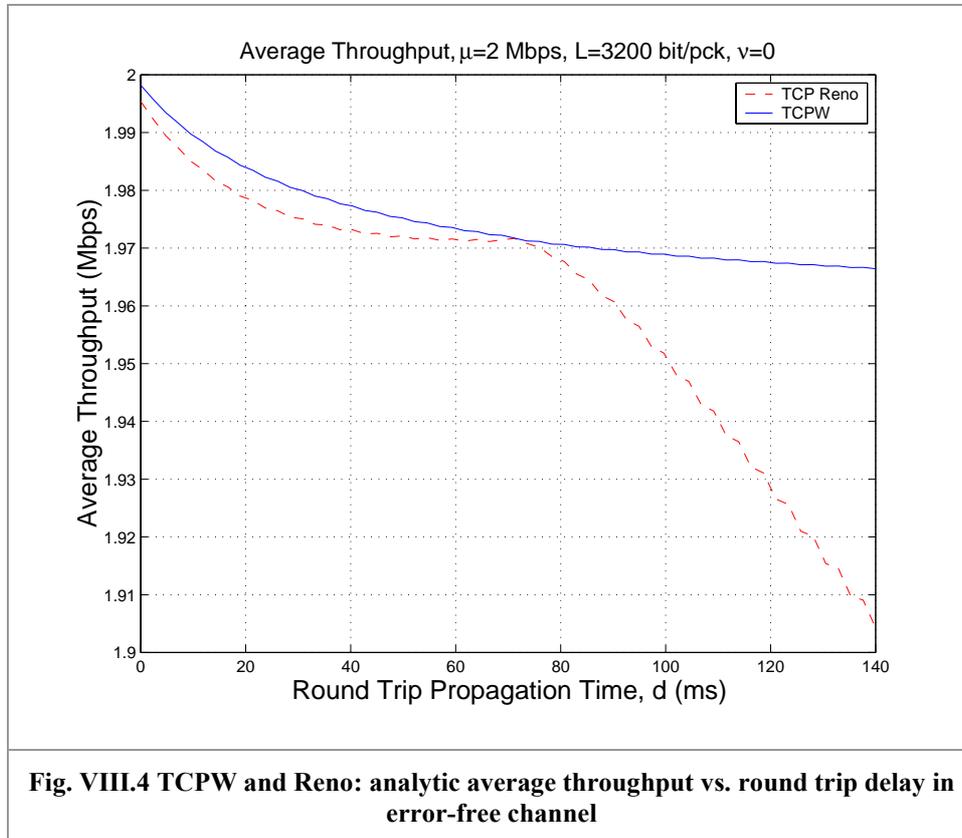


Fig. VIII.3 shows the average throughput of TCPW and Reno with respect to variations of the bottleneck speed. In this case, the buffer size has been kept equal to the nominal pipe capacity ( $B=44$  pcks). For link bandwidth less than the nominal value ( $\mu=2$  Mbps), the two protocols exhibit the same performance. As the link bandwidth increases, the measurement-based nature of TCPW allows it to track the bandwidth variations of the bottleneck and to linearly build-up its performance; TCP Reno also improves its performance but in a less considerable way.

Finally, Fig. VIII.4 shows the behavior of the average throughput achieved by TCPW and Reno against the round trip delay. From the figure it is clear that TCPW looks much more robust with respect to variations of RTT. In particular, it is worth pointing out that

the Reno curve touches the TCPW curve for the nominal value of  $RTT=70$  msec, for which the buffer size is equal to the pipe size.



### VIII.5. A simplified analytic model

The analytic model we developed so far takes into account all the information the protocol needs in its dynamic behavior. Unfortunately, such a highly detailed description makes the model somewhat cumbersome. On the other hand, for many purposes a simpler though slightly approximate model would be sufficient. In this section, on the basis of a reasonable approximation, we will derive a simpler model of TCPW behavior and show that the performance results we achieve match closely the ones obtained from the detailed model.

The major drawback of the detailed model is the lack of a closed form expression for the bandwidth estimated by the low pass filter. This is due to the irregular bandwidth samples the filter receives at the beginning of every burst. In the following we postulate that those samples do not significantly affect the estimated bandwidth. We assume that

all samples are equal to  $\mu$  except the first one in the cycle, which equals  $1/RTT$ . The input process of the filter can then be simplified as follows:

$$\tilde{b}_n = \begin{cases} 1/RTT & n = 0 \\ \mu & \text{elsewhere} \end{cases} \quad (\text{VIII.21})$$

The solution of (VIII.13) is then straightforward for  $n \geq 3$  and is given by:

$$BWE(n) = \mu - (\mu - BWE(2))\alpha^{n-2}, \quad (\text{VIII.22})$$

while  $BWE(0) = W_0/RTT$  and the values of  $BWE(1)$  and  $BWE(2)$  can be easily derived through the recursion (VIII.13).

Formula (VIII.22) can be successfully applied to the evaluation of the transition probabilities  $P_{i,j}$ . The transition expressed as  $\{W_0^{(m)} = j | W_0^{(m-1)} = i\}$  can be re-written in terms of the estimated bandwidth as:

$$\left\{ \left[ RTT \cdot BWE(N_{drop}) \right] = j \mid BWE(0) = \frac{i}{RTT} \right\} \quad (\text{VIII.23})$$

and thus:

$$\left\{ \left[ \frac{j}{RTT} \right] \leq BWE(N_{drop}) < \left[ \frac{j}{RTT} \right] + 1 \mid BWE(0) = \frac{i}{RTT} \right\} \quad (\text{VIII.24})$$

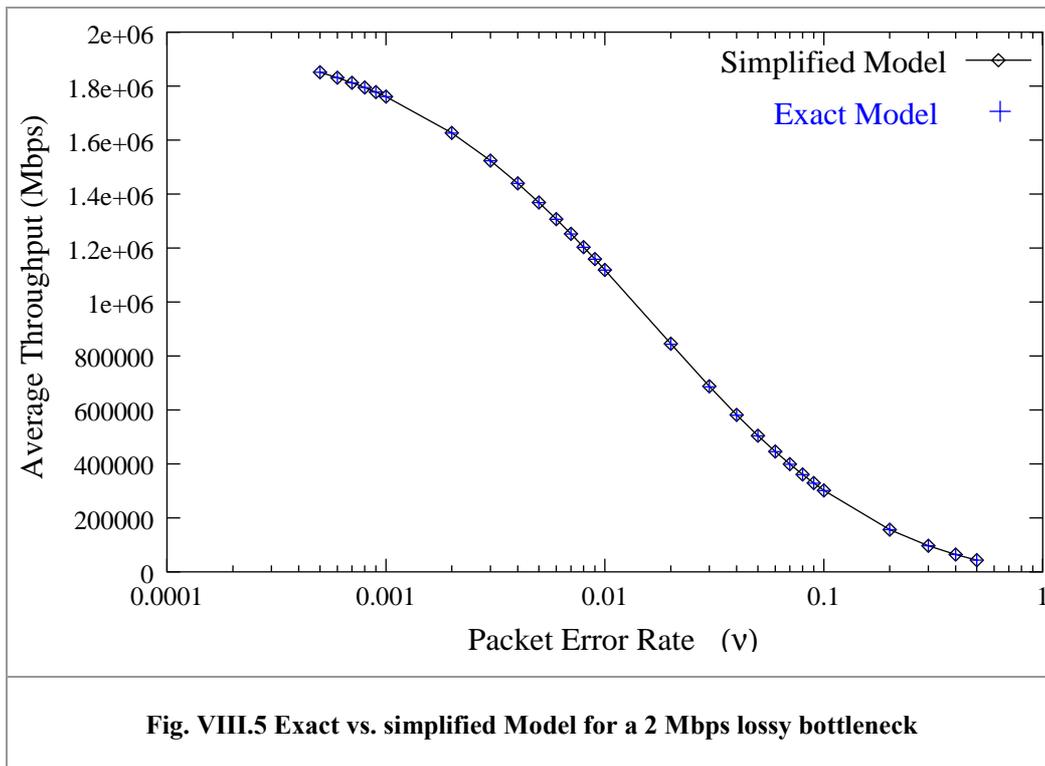
Now, the relation in (VIII.24) can be solved exactly with respect to  $N_{drop}$ , obtaining as a result the set  $[n_l, n_u)$ . For the sake of conciseness we omit here the trivial but cumbersome explicit computation of  $n_l$  and  $n_u$  as they may assume different value when either or both are less than 3. The transition probabilities  $P_{i,j}$  are then given by:

$$P_{i,j} = \Pr \left[ W_0^{(m)} = j \mid W_0^{(m-1)} = i \right] = \sum_{n=n_l}^{n_u-1} p_{i,n} \quad (\text{VIII.25})$$

where  $p_{i,n}$  is given by (VIII.17).

Once the transition probability matrix has been obtained, the procedure to get the throughput follows the one presented in subsection VIII.3.B. Fig. VIII.5 shows the

results obtained using the simplified models compared to the ones obtained from the complete model. From a quick visual inspection it is easy to notice that the two approaches lead to very close results, while the simplified technique is much faster in calculating the transition matrix probabilities.



## VIII.6. Conclusion

In this chapter we developed an analytic model for the TCP Westwood (TCPW) protocol. TCPW is a new TCP scheme, which requires modifications only in the TCP source stack and is thus compatible with TCP Reno and Tahoe destinations. Basically it differs from Reno in that it adjusts the *cwnd* (congestion window) after a loss detection by setting it to the *measured rate* currently experienced by the connection, rather than using the conventional multiplicative decrease scheme.

The analytic model was developed in order to study the behavior of TCPW in presence of random errors and under different scenarios. It also represents a useful contribution in that it provides further insight in TCPW operation and allows cross validation against simulation and measurements results.

Following previous works on TCP Reno, we described the evolution of the congestion window by means of a Markov chain. However, unlike previous analytic models, the TCPW model must address the bandwidth estimation mechanism and its impact on system behavior. The model was validated by means of simulation in the general case of a lossy link. Then we used it to derive TCPW performance in the ideal error-free channel.

The results show that, for a single connection case, TCPW protocol performs better than or, at least, as well as TCP Reno in terms of average throughput. The results also show that TCPW is more robust under varying buffer size, round trip delays and bottleneck bandwidth.

---

## IX. CONCLUSIONS

---

The thesis has addressed the analysis and modeling of personal and local wireless data networks. The aim of the research was to gain a deeper understanding of the problems related to such systems, providing new instruments of analysis and innovative solutions. To reach this goal, we have investigated several different aspects concerning wireless communications, ranging from the physical layer, to the medium access control, up to the transport layer.

The study has focused, first, on the emerging Bluetooth radio technology, which has been proposed as standard wireless interface for Personal Area Networks (PANs). Foremost, we have investigated the performance achieved over a point-to-point radio link by the six different data radio packet formats supplied by Bluetooth. For this purpose, we have performed a large series of experiments, consisting in heavy file transfers over a Bluetooth asynchronous connection less (ACL) link, and we have collected the transmission statistics. The experimental approach has given us an insight of the system behavior in real operating conditions. Measurements have shown that, packet length being equal, unprotected data formats (DHn) achieve higher performance than protected formats (DMn). The forward error correction (FEC) code used by DMn packets is actually able to correct most of errors in the payload field. Nevertheless, the improvement in packet error probability does not compensate for the inefficiency of FEC overhead, unless for very critic channel conditions.

Furthermore, long packets have shown better performance than short ones in almost all the situations that we have considered. Although such a result was partially predictable, since longer packet formats have greater payload capacity, a tradeoff between long and short packet formats could have been expected. Indeed, longer is the packet higher should have been the payload error probability. Data has shown, however, that the three unprotected packet formats, DH1, DH3 and DH5, experiment roughly the same payload error probability, independently of their length. This observation has suggested to adopt a three-state Markov model for modeling the Bluetooth radio link, in

which the states correspond to the following three reception events: synchronization failure (access code or packet header not recoverable), packet received with errors, and packet received with no error. Performance metrics have been analytically derived from the model and compared to the experimental results, showing a good correspondence. The model captures the aggregate of real-world effects like fading, interference and noise and, hence, it appears suitable to be utilized for modeling the Bluetooth radio link behavior in simulating platforms.

Successively, we have considered issues concerning medium access control strategies for Bluetooth piconets and scatternets. In this context, we have pointed out the relationship of duality that ties the roles of masters and gateways in the scatternet. On the basis of such a duality, we have extended the concept of max-min fair share to a scatternet topology, highlighting the *forces* that act on the gateways and determine the fair share of their time among the piconets they belong to. Furthermore, we have proposed a totally distributed scatternet scheduling algorithm that provides an integrated solution for both intra- and inter-piconet scheduling. A simple analysis of the algorithm has proven its capability of achieving max-min fair resource distribution in piconets. Following the duality principle, the algorithm has been then adopted even for schedule the gateways presence among piconets in a max-min fair manner. Computer simulations have shown that the algorithm adapts to non-uniform and changing traffic, without requiring any exchange of control information, and tends to achieve max-min fairness all over the scatternet.

The study has, then, considered the internetworking of isolated Bluetooth scatternets by means of other radio technologies with higher coverage range. In particular, we have proposed an hybrid architecture of UMTS/TDD and Bluetooth that can exploit the low-cost advantage of Bluetooth without the limitations of its small range. The analysis has been focused on a specific centralized topology, in which traffic flows between a central unit and many peripheral nodes only. In such an architecture, Bluetooth provides network access interface to the final users, by means of strategically positioned Bluetooth access points. The access points are wirelessly interconnected in static scatternets. Each scatternet contains a hybrid unit, equipped with both Bluetooth and UMTS interfaces, that concentrates the traffic from and to the peripheral units and

bridges the scatternet to a UMTS base station. Though particular, such a centralized topology may be applied in many different contexts where the cost-effectiveness of the network is more important than its performance. For the proposed architecture, we have considered the problem of scheduling the presence of the gateways in the piconets in a fair manner. The analysis has shown that, in general, the system capacity can be fairly distributed among users, unless they are heavily concentrated in one section of the network.

Successively, we have considered the medium access control problems for Local Multipoint Distribution Services (LMDS) systems. In such systems, a central unit provides a large number of digital services to many subscribers, scattered in a limited area. A point-to-multipoint radio channel is utilized to carry the downlink traffic (from service provider to final users). Uplink communication is, instead, provided by means of a common, point-to-point channel. The analysis has focused on the Contention-Time Division Multiple Access (C-TDMA) protocol, which has been proposed to manage the multiuser access to the uplink channel. The protocol is based on a TDMA scheme with slot reservation. In order to get a reservation, users have to win a contention phase. At the beginning of each frame, users randomly choose an idle slot in the frame and transmit a packet on it. If the transmission is successful the slot is marked as “reserved” by the base station, and transmission can take place on that slot in the following frames. On the contrary, the user remains in the contention phase.

C-TDMA has been studied by using both classical Markov analysis and a simplified tool known as equilibrium point analysis (EPA), which for practical conditions (large number of users and channels) is more convenient, due to its reduced computational cost. Performance evaluation of C-TDMA has been made in terms of throughput and delay by using EPA and computer simulations. It has also been proposed a method of optimization of C-TDMA design to improve the maximum achievable throughput. The results indicate that C-TDMA yields high throughput values with a limited delay in typical cellular scenarios. However, as observed for other access protocols [15][58], the C-TDMA can show, in particular cases, a bistable behavior, in which periods of high performance alternate with periods of block. A comparison with the Packet Reservation Multiple Access (PRMA) protocol has been made, in order to have a benchmark for the

performance achieved by the C-TDMA. The comparison has been carried out both in a data transmission scenario without real time constraints and in a voice scenario with packet dropping. Results have shown that the protocols achieve very similar performance in terms of throughput, delay and packet dropping probability in both the scenarios that have been considered.

Finally, we have investigated the problems concerning the transmission of TCP/IP protocols over wireless networks. Radio channels usually present time varying behavior, with long delay and burst of errors. These characteristics do not combine with TCP, which has been tuned to work properly in wired networks, where segment loss and long delay are mainly due to congestion. Thus, the transmission of TCP over a wireless connection may lead to heavy performance losses, since TCP may invoke congestion reaction algorithms even in absence of congestion, misinterpreting the effects due to the radio channel.

In the thesis, we have analyzed a new version of the TCP protocol, named TCP Westwood (TCPW), which aims to alleviate the problem of sporadic segment losses by accelerating the successive recovery phase. The protocol constantly estimates the bandwidth available through the connection, by using the arrival process of acknowledgements at the sender side. The bandwidth estimation is then used to reset the congestion window, after a segment loss event, to an appropriate value.

We have proposed an analytical model for TCPW protocol, which has been validated through computer simulations. Following the approach suggested by previous works on TCP Reno, we have described the evolution of the congestion window by means of a Markov chain. In the case of TCPW, however, we have included in the model the bandwidth estimation mechanism, an element that was not present in the previous works. The model has been developed in order to investigate the performance achieved by the TCPW protocol in different scenarios. The results have shown that, for a single connection case, TCPW protocol performs better than or, at least, as well as TCP Reno in terms of average throughput. Furthermore, TCPW has proved to be more robust under varying buffer size, round trip delays and bottleneck bandwidth.

## IX.1. List of published papers

Some of the results here presented were collected in technical papers, which have been submitted to international conferences, magazines and journals. At this time, we have received notification of acceptance for the following publications:

- G.Pierobon, A.Salloum, A. Zanella, "Contention-TDMA protocol: Performance Evaluation," to appearing in IEEE trans. on Vehicular Technology, Jan. 2002.
- A.Zanella, D.Melpignano, "Analysis of File Transfer Protocol Over Bluetooth Radio Link", 12th Tyrrhenian International Workshop on Digital Communications "Software Radio Technologies and Services", Portoferraio - Isle of Elba (Italy), Sept. 2000.
- M. Gerla, A. Zanella, "Bluetooth: una nuova tecnologia per reti radio personali," Notiziario Tecnico di Telecomitalia, Settembre 2001 (in Italian).
- A. Zanella, G. Procissi, M. Gerla, M.Y. Sanadidi, "TCP westwood: analytic model and performance evaluation," Proceeding of Globecom 2001, S. Antonio, Texas, dicembre 2001.
- M. Gerla, M.Y. Sanadidi, R. Wang, A. Zanella, C. Casetti, S. Mascolo, "TCP Westwood: congestion window control using bandwidth estimation," Proceeding of Globecom 2001, S. Antonio, Texas, dicembre 2001.
- T. Kwon, A. Zanella, R. Kapoor, Y. Lee, M. Gerla, "A hybrid architecture of UMTS and Bluetooth for indoor wireless/mobile communications," Proceeding of ICWLHN 2001, Singapore, dicembre 2001.
- M. Gerla, M.Y. Sanadidi, R. Wang, A. Zanella, C. Casetti, S. Mascolo, "TCP Westwood: bandwidth driven TCP window control," Poster session at DARPA Meeting April 2-4, 2001 La Jolla, CA



---

## X. REFERENCES

---

- [1] "A wireless LAN access point." *Microwave Journal, Euro-Global Edition*, vol.40, (no.1), Horizon House Publications, Jan. 1997.
- [2] "Radio Equipment and Systems (RES): High Performance Radio Local Area Network (HIPERLAN), Type 1 functional specification," European Telecommunications Standards Institute, DRAFT PE6 prETS 300 652, Paris, Dec. 1995.
- [3] Abouzeid A., Roy S., Azizoglu M., "Stochastic Modeling of TCP over Lossy Links," *INFOCOM2000*
- [4] Allman M. and Paxson V., "On Estimating End-to-End Network Path Properties", *SIGCOMM 1999*.
- [5] Asatani K. and Maeda Y., "Access network architectural issues for future telecommunications systems," *IEEE Commun. Mag.*, pp. 110-114, Aug. 1998.
- [6] Augé A. C., Aspás J. P., "TCP/IP over Wireless Links: Performance Evaluation", *VTC '98*.
- [7] Bakre A. and Badrinath B. R., "I-TCP: Indirect TCP for Mobile Hosts". In *Proc. 15th International Conf. on Distributed Computing Systems (ICDCS)*, May 1995.
- [8] Balakrishnan H. and Katz R. H., "Explicit Loss Notification and Wireless Web Performance," In *Proceedings of IEEE GLOBECOM'98 Internet Mini-Conference*, Sydney, Australia, November 1998.
- [9] Balakrishnan H., Padmanabhan V. N., Seshan S., and Katz R. H., "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links," *IEEE/ACM Transactions on Networking*, December 1997.
- [10] Balakrishnan H., Seshan S., Amir E., and Katz R. H., "Improving TCP/IP Performance Over Wireless Networks," *MOBICOM'95*, Berkeley, CA, USA, November 1995
- [11] Balakrishnan H., Seshan S., and Katz R.H., "Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks". *ACM Wireless Networks*.
- [12] Benvenuto N., Salloum A., Tomba L., "Performance of digital DECT radio links based on semi-analytical methods", *IEEE J. Selected Areas in Comm.*, vol.15, no. 4, May 1997.

- [13] Bruno R., Conti M., Gregori E., “Wireless Access to Internet via Bluetooth: Performance Evaluation of the EDC Scheduling Algorithm”, The First Workshop on Wireless Mobile Internet, July 21, 2001, Rome, Italy.
- [14] Capone A., R. Kapoor and M. Gerla: “Efficient Polling Schemes for Bluetooth Picocells”, ICC 2001
- [15] Carleial A. and Hellman M., “Bistable behavior of ALOHA-type systems,” IEEE Transaction on Communication, vol. COM-23, pp. 401-409, Apr. 1975.
- [16] Casetti C., Gerla M., Lee S., Mascolo S., and Sanadidi M., “TCP with Faster Recovery,” MILCOM 2000, Los Angeles, CA, October 2000.
- [17] Cerf V. C. and Kahn R. E., “A Protocol for packet Network Interconnections,” IEEE Transactions on Communications, vol. COM-22, no. 5, pp. 637-648, May 1974.
- [18] Chan A., Tsang D. and Gupta S., “TCP (Transmission Control Protocol) over Wireless Links”, Proc. of VTC'97, Phoenix, USA. May 1997.
- [19] Chiasserini C.F, Rao R.R, “Performance of IEEE 802.11 WLANs in a Bluetooth environment”, Wireless Communications and Networking Conference, WCNC 2000
- [20] Chockalingam A., Zorzi M. and Tralli V., “Wireless TCP Performance with Link Layer FEC/ARQ,” Proc. of International Conference on Communications, Vancouver, Canada, June 1999.
- [21] Das A., Ghose A., Razdan A., Saran H. & Shorey R., “Enhancing Performance of Asynchronous Data Traffic over the Bluetooth Wireless Ad-Hoc Network,” Proc. IEEE VTC 2000, Tokyo (Japan), May 2000.
- [22] DeSimone A., Chuah M. C., and Yue O. C., “Throughput Performance of Transport-Layer Protocols over Wireless LANs”. In Proc. Globecom '93, December 1993.
- [23] Digital Video Broadcasting (DVB); DVB interaction channel for LMDS distribution system. European Telecommunications Standards Institute. Working Draft prETS 300. Jun. 1997.
- [24] Elliot O. L., “Estimates of error rates for codes on burst-noise channels,” The Bell System Technical Journal, September 1963.
- [25] ETSI RES, “Digital European Cordless Telecommunications (DECT), Common interface Part 1: Overview,” ETS 300 175-1, 1996.
- [26] ETSI TR 125 922 V3.5.0 (2001-03) Universal Mobile Telecommunications System (UMTS), [http://www.3gpp.org/3G\\_Specs/3G\\_Specs.htm](http://www.3gpp.org/3G_Specs/3G_Specs.htm).

- 
- [27] Fabian O., Levy H., “Polling System Optimization Through Dynamic Routing Policies,” IEEE INFOCOM ’93, vol. 1, pp. 194–200.
- [28] Fritchman B. D., “A binary channel characterization using partitioned Markov chain,” IEEE Trans. on Information Theory, pp. 221–227, April 1963.
- [29] Garg S., Kalia M., Shorey R., “MAC Scheduling Policies and SAR Policies for Bluetooth: A Master Driven TDD Pico–Cellular Wireless System,” MoMuc 99, pp. 384–386
- [30] Gessner C. and Köhn R., Schniedenharn J., and Sitte A., “Layer 2 and Layer 3 of UMTS-TDD,” IEEE VTC 2000 Spring, pp. 1181 - 1185, Tokyo, 2000.
- [31] Gilb, J.P.K “Bluetooth radio architectures.” 2000 IEEE Radio Frequency Integrated Circuits (RFIC) Symposium Digest of Papers, Boston, MA, USA, 11–13 June 2000
- [32] Gilbert E. N., “Capacity of a burst–noise channel,” The Bell System Technical Journal, September 1960.
- [33] Goff T., Moronski J., Phatak D. S., and Gupta V., “Freeze–TCP: a True End–to–end TCP Enhancement Mechanism for Mobile Environments,” In Proceedings of IEEE INFOCOM’2000, Tel Aviv, Israel, March 2000.
- [34] Goodman D., Valenzuela R., Gayliard K., and Ramamurthi B., “Packet reservation multiple access for local wireless communication,” IEEE Transaction on Communication, vol. 37, pp. 885–890, Aug. 1989.
- [35] Haartsen, J.C. ‘Bluetooth towards ubiquitous wireless connectivity.’, Revue HF, Soc. Belge Ing. Telecommun. & Electron, 2000. p.8–16.
- [36] Haartsen, J.C. “The Bluetooth radio system.”, IEEE Personal Communications, IEEE, Feb. 2000.
- [37] Haartsen, J.C.; Mattisson, S. “Bluetooth—a new low–power radio interface providing short–range connectivity.” Proceedings of the IEEE, IEEE, Oct. 2000.
- [38] Hoe J. C., “Improving the Start–up of A Congestion Control Scheme for TCP”, Proc. ACM SIGCOMM ’96, pp. 270–280.
- [39] Hoe J. C., “Start-up Dynamics of TCP’s Congestion Control and Avoidance Schemes”. Master’s thesis, Massachusetts Institute of Technology, 1995.
- [40] Jacobson V., “Congestion Avoidance and Control,” ACM Computer Communications Review, 18(4) : 314 - 329, August 1988.
- [41] Jacobson V., “Modified TCP Congestion Avoidance Algorithm”, end2end-interest mailing list, April 30, 1990. <ftp://ftp.isi.edu/end2end/end2end-interest-1990.mail>.

- [42] Jinsong Wu; Ilow, J. "A wireless multimedia LAN testbed," 2000 Canadian Conference on Electrical and Computer Engineering, Halifax, NS, Canada, 7–10 March 2000.
- [43] Johansson N., Korner U., Johansson P., "Performance Evaluation of Scheduling Algorithms for Bluetooth", In *Broadband Communications: Convergence of Network Technologies*, Edited by Danny H. K. Tsang and Paul J. Kuhn
- [44] Johansson P., Kazantzidis M., Kapoor R., Gerla M., "Bluetooth – an Enabler for Personal Area Networking", *IEEE Network Magazine, Wireless Personal Area Network*, September 2001.
- [45] Kalampoukas L., Varma A., and Ramakrishnan K. K., "Explicit Window Adaptation: A Method to Enhance TCP Performance," In *Proceedings of IEEE INFOCOM'98*, San Francisco, Ca, USA, March/April 1998.
- [46] Kalia M., Bansal D., Shorey R., Data scheduling and SAR for bluetooth MAC, *IEEE VTC 2000–Spring Tokyo*, pp–716–720
- [47] Kamerman, A.; Monteban, L. "WaveLAN–II: a high–performance wireless LAN for the unlicensed band." *Bell Labs Technical Journal*, vol.2, Lucent Technologies, Summer 1997.
- [48] Kleinrock L. and Lam S., "Packet switching in a multiaccess broadcast channel: performance evaluation," *IEEE Transaction on Communication*, vol. COM–23, pp. 410–422 , Apr. 1975.
- [49] Lai Kevin and Baker Mary, "Measuring Link Bandwidths Using a Deterministic Model of Packet Delay", *SIGCOMM 2000*
- [50] Lakshman T. and Madhow U., "The performance of TCP/IP for networks with high bandwidth-delay products and random loss," *IEEE/ACM Transactions on Networking*, vol. 5, no. 3, 1997.
- [51] Lam S.S., "Packet broadcast networks- A performance analysis of the R–ALOHA protocol," *IEEE Trans. Comput.*, vol. COM–29, pp. 596–603, Jul. 1980.
- [52] Li S. Q. and Hwang C., "Link Capacity Allocation and Network Control by Filtered Input Rate in High speed Networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 1, pp. 10 - 25, Feb. 1995.
- [53] Liu Z., Nain P., Towsley D., "On Optimal Polling Policies," *Queueing Systems*, Vol. 11, pp. 14141–1420.
- [54] Mayer A., Ofek Y., Yung M., "Approximating Max–Min fair Rates via Distributed Local Scheduling with Partial Information", In the proceedings of *IEEE INFOCOM 1996*.

- 
- [55] Mihailescu C., Lagrange X., and Godlewski Ph., “Dynamic Resource Allocation for Packet Transmission in UMTS TDD TD-CDMA Systems”, IEEE Vehicular Technology Conference (VTC), Houston, USA, May 1999.
- [56] Mouly M. and Pautet M.-B., *The GSM System for Mobile Communications*, 1992.
- [57] Nagle J., “Congestion Control in IP/TCP Internetworks”, RFC 896, Jan.1984.
- [58] Nanda S., “Stability evaluation and design of the PRMA joint voice data system,” IEEE Transaction on Communication, vol. 42, pp. 2092–2104, May 1994.
- [59] Nanda S., Goodman D. and Timor U., “Performance of PRMA: a packet voice protocol for cellular systems,” IEEE Transaction on Vehicular Technologies, vol. 40, pp. 584–598, Aug. 1991.
- [60] Pahlavan K. and Levesque A. H., “Wireless Information Networks,” JOHN WILEY & SONS, INC. Publication.
- [61] Perkins C., Royer E., “Ad-hoc on-demand distance vector routing”, *Mobile Computing Systems and Applications*, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on , 1999 , Page(s): 90 –100.
- [62] Proakis J., “Digital Communications.” Third Edition, McGraw–Hill, New York, 1995.
- [63] Rathi S., “Bluetooth protocol architecture.” *Dedicated Systems Magazine*, *Dedicated Systems Experts*, Oct.–Dec. 2000.
- [64] Riordan J., “An introduction to combinatorial analysis,” Princeton University Press, New Jersey, 1978.
- [65] Salloum A., Benvenuto N., Coppola G. and Pierobon G., “A DECT–based return channel for 42 GHz broadband wireless systems,” *Electronics Letters*, vol. 34, pp. 945–946, May 1998.
- [66] Salloum A., Benvenuto N., Coppola G. and Pierobon G., “Access protocols for cellular high–speed data services,” *Int. J. of Wireless Inform. Networks*
- [67] Salloum A., Benvenuto N., Coppola G. and Pierobon G., “Uplink access protocol for multimedia systems based on DECT/ATM Layer,” *VTC'98*, Ottawa, Canada, pp. 117–121, May 18–21, 1998.
- [68] Salonidis T., Bhagwat P., Tassioulas L., LaMaire R., “Distributed Topology Construction of Bluetooth Personal Area Networks”, In the proceedings of IEEE INFOCOM 2001, Anchorage, Alaska, USA, April 22–26, 2001.

- [69] Siep, T.M.; Gifford, I.C.; Braley, R.C.; Heile, R.F., "Paving the way for personal area network standards: an overview of the IEEE P802.15 Working Group for Wireless Personal Area Networks." IEEE Personal Communications, vol.7, (no.1), IEEE, Feb. 2000.
- [70] Specification of Bluetooth System, ver. 1.0, July 1999
- [71] Specifications of IEEE 802.11 <http://standards.ieee.org/getieee802/802.11.html>
- [72] Stevens W. R., "TCP/IP Illustrated, Volume 1: The Protocols", Addison-Wesley, 1994.
- [73] Stevens W. R., Wright G. R., "TCP/IP Illustrated, Volume 2: The Implementation", Addison-Wesley, 1995.
- [74] Stevens W., "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms", RFC 2001Jan.1997.
- [75] Stevens W., Allman M., "TCP Congestion Control", RFC 2581 Apr. 1999.
- [76] Stüber G.L., "Principles of mobile communication," Boston, Kluwer Academic, 1996
- [77] Swarts J. and Ferreira H., "On the evaluation and application of Markov channel model in wireless communications," VTC '99.
- [78] Takai M. L., Ahuja B.R., Bagrodia R. and Gerla M., "GloMoSim: A Scalable Network Simulation environment," Technical report 990027, UCLA, Computer Science Department, 1999.
- [79] Tanenbaum A.S., "Computer Networks", Third Edition, Prentice-Hall International, Inc.
- [80] Tasaka S., "Stability and performance of the R-ALOHA packet broadcast system," IEEE Trans. Comput., vol. C-32, pp. 717-726, Aug. 1983.
- [81] The Bluetooth Special Interest Group, <http://www.bluetooth.com/>
- [82] TIA/EIA IS-95B, "Mobile Station-Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular Systems," 1998.
- [83] TIA/EIA/IS-136.2, "800 MHz TDMA Cellular-Radio Interface-Mobile Station-Base Station Compatibility — Traffic Channels and FSK Control Channel," Dec. 1994.
- [84] Wang H. S. and Moayeri N., "Finite-state Markov channel: A useful model for radio communication channels," IEEE Trans. on Vehicular Technology, pp.163-171, February 1995.

- [85] Wang H. S., “On verifying the first–order Markovian assumption for a Rayleigh fading channel model,” *IEEE Trans. on Vehicular Technology*, pp.353–357, May 1996.
- [86] Wittenmark K. J. B., “Computer controlled systems,” Prentice Hall, Englewood Cliffs, N. J., 1997. ns–2 network simulator (ver 2). LBL, URL: <http://www-mash.cs.berkeley.edu/ns>.
- [87] Yaiz R. A., Heijenk G., “Polling in Bluetooth: a simplified best effort case,” *Proceedings CTIT Workshop on Mobile Communications*, 2000.
- [88] Zeng X., Bagrodia R., and Gerla M., “GloMoSim: a library for parallel simulation of large-scale wireless networks”, In *Proceedings of the 12th Workshop on Parallel and Distributed Simulations*, May 1998.
- [89] Zhang L., Shenker S., and Clark D.D., “Observations on the Dynamics of a Congestion Control Algorithm: The Effects of Two-Way Traffic”. *Proc. of ACM SIGCOMM '91*, pages 133-147, Sep. 1991.