TABLE OF CONTENTS

Abstract i

Acknowledgements ii

Table of Contents iii

List of Figures ix

List of tables xii

Chapter 1 1

1. Introduction and Preview 1

1.1. Motivation and challenges 1

1.2. Thesis outline 2

Chapter 2 4

2. An Introduction to Ad Hoc Mobile Networks 4

2.1. Introduction 4

2.2. Wired vs. Wireless networking 5

2.3. Wireless evolution 6

2.4. Ad Hoc Mobile Networks 8

2.5. Characteristics of ad hoc mobile networks 9

2.5.1. Characteristics due to distribution 9

2.5.2. Characteristics due to wireless links nature 9

2.5.3. Mobility 10

2.5.4. Multi hopping 11

2.5.5. Limited energy source 11

2.6. Ad hoc mobile networks applications 11

2.7. Conclusion 12

Chapter 3 13

3. Unicast Routing in Ad Hoc Wireless Networks 13

3.1. Introduction 13
3.2. Routing in conventional networks 14
3.3. Routing in ad hoc mobile networks 15
3.4. Classifications of routing protocols 16
3.4.1. Proactive VS On–Demand routing protocols 17
3.4.2. Distributed VS source routing 17
3.4.3. Flat VS Hierarchical routing 17
3.4.4. Geographical approaches 17
3.5. Proactive routing protocols 18
3.5.1. Destination–sequenced distance vector protocol (DSDV) 18
3.5.2. Optimized link state routing protocol (OLSR) 20
3.5.3. Topology dissemination based on reverse–path forwarding (TBRPF) [25] 22
3.6. Reactive routing protocols 25
3.6.1. Temporally Ordered Routing algorithm (TORA) 25
3.6.2. Dynamic source routing 32
3.6.3. Ad hoc on–demand distance vector routing protocol 36
3.7. Conclusion 38

Chapter 4 39

4. MAC Layer Protocols for Ad Hoc Mobile Networks 39

4.1. Introduction 39
4.2. The multiple access to wireless mediums 40
4.2.1. Time division multiple access 40
4.2.2. Frequency division multiplexing 41
4.2.3. Code division multiplexing 42
4.2.4. Space division multiplexing 42
4.3. Ad hoc key considerations for Mac protocols design 43
4.3.1. A shared and multi–access medium 43
4.3.2. Varying channel condition 43
4.3.3. Errors ratio 44

4.3.4. Signals properties (speak XOR hear) 44
4.3.5. Hidden terminal and exposed terminal problems 45
4.4. Design goals for MAC protocols in MANETs 45
4.4.1. Controlled latency 46
4.4.2. Throughput and goodput 46
4.4.3. Fairness 46
4.4.4. Controlled overhead 46
4.4.5. Power efficiency 46
4.4.6. QoS delivery 47
4.4.7. The physical layer facilities 47
4.5. Review of MAC protocols for ad hoc networks 47
4.5.1. ALOHA [45] 48
4.5.2. Slotted ALOHA [46] 49
4.5.3. Carrier Sense Multiple Access 50
4.5.4. Elimination yield non–preemptive priority multiple access (EY–NPMA) [47] 51
4.5.5. Multiple Access Collision Avoidance (MACA) [49] 53
4.5.6. MACA for Wireless (MACAW) [50] 54
4.5.7. MACA by invitation (MACA–BI) [51] 55
4.5.8. Dual Busy Tone Multiple Access (DBTMA) [52] 55
4.5.9. Floor Acquisition Multiple Access (FAMA/FAMA–NCS [56, 57]) 56
4.5.10. The distributed coordination function of the IEEE 802.11 (DCF) [59] 56
4.6. Conclusion 58

Chapter 5 59

5. QoS Support in Ad Hoc Mobile Network 59

5.1. Introduction 59
5.2. About Quality of Service 60
5.3. QoS in the Internet 61
5.3.1. The Integrated Services model (IntServ) 61
5.3.2. Differentiated service 63
5.4. QoS in MANETs 63
5.5. General models for QoS support in MANETs 64
5.5.1. IntServ and DiffServ in ad hoc mobile networks 64

5.5.2. Flexible QoS Model for MANETs (FQMM) [65] 65
5.5.3. Stateless Wireless Ad hoc Networks (SWAN) [66] 66
5.5.4. INSIGNIA [68] 68
5.6. QoS support from a layered perspective 72
5.6.1. QoS provisioning at the MAC layer 72
5.6.2. QoS routing 76
5.6.3. Transport layer in MANETs and QoS 82
5.6.4. Application layer 82
5.7. Conclusion 83

Chapter 6 84

6. Cross Layer Design of Ad Hoc Mobile Networks 84

6.1. Introduction 84
6.2. What is CLD? 85
6.3. Motivation for CLD 86
6.4. The cost of cross–layer optimizations 87
6.4.1. Design, implementation and modeling complexity 88
6.4.2. Interoperability with existing systems and coexistence of multiple CLDs 88
6.4.3. The system longevity 89
6.5. Cross–layer design proposals: a taxonomy 89
6.5.1. Cross–layering based on added notifications 90
6.5.2. Cross–layering based on added interactions 90
6.5.3. Merging adjacent layers 91
6.5.4. Design coupling 91
6.5.5. Cross–layering by adding an information sharing mechanism 92
6.6. Examples involving cross layer 92
6.6.1. Cross–layer congestion control 92
6.6.2. MOBILEMAN 95
6.6.3. CrossTalk 96
6.7. Performance of cross–layer designs: a bad example 96
6.7.1. Rate adaptive MAC with minimum hop routing 96
6.8. Conclusion 97

Chapter 7 99

7. EFORTS Every node Feedback for Optimizing Real time Traffic Support 99

7.1. Introduction 99
7.2. Why VoIP? 100
7.3. VoIP transmission 100
7.3.1. Digitization 101
7.3.2. Speech coding 101
7.4. Quality of VoIP 104
7.4.1. End–to–end delay 105
7.4.2. Jitter 107
7.4.3. Loss 108
7.4.4. Voice quality measurement 109
7.5. Real time protocols, RTP & RTCP 110
7.6. Signaling protocols 111
7.6.1. SIP 111
7.7. VoIP in ad hoc networks 112
7.7.1. New challenges 112
7.7.2. Convenient directions 113
7.7.3. Permissive error control 113
7.7.4. Design of specific speech coders 113
7.7.5. Renew network protocols design 113
7.8. EFORTS (Every node Feedback for Optimizing Real time Traffic Support) 114
7.9. Founding our proposal 114
7.9.1. 1st Ad Hoc networks are poor environments 114
7.9.2. 2nd Ad Hoc networks are highly dynamic 115
7.9.3. 3rd Ad Hoc networks offer new opportunities 116
7.10. The proposed scheme (EFORTS) 116
7.10.1. Packets deadline aware MAC 117
7.10.2. The MAC Spy 118
7.10.3. The reporting agent RA 118
7.10.4. The state collector SC 120

7.10.5. RA interaction with routing 121
7.10.6. RA interaction with adaptive applications 121
7.10.7. RA and admission control 121
7.11. Simulations 122
7.11.1. VoIP support in unloaded MANETs 122
7.11.2. A more realistic scenario 132
7.12. Conclusion 140

Chapter 8 141

8. Conclusions and Future Work 141

8.1. Conclusions 141
8.2. Future work 142

References 143

LIST OF FIGURES

Figure 3-1: MPRs selection. 21
Figure 3-2: Topology information propagation in TBRPF. 24
Figure 3-3: Destination oriented DAG. 26
Figure 3-4: Routes creation in TORA. 29
Figure 3-5: Maintaining routes in TORA. 31
Figure 3-6: Routes discovery in DSR. 34
Figure 3-7: Resending route error packets by the originator DSR. 36
Figure 3-8: Routes creation in AODV. 37
Figure 4-1: Time Division Multiple Access. 41
Figure 4-2: Frequency Division Multiple Access. 41
Figure 4-3: Space Division Multiple Access. 42
Figure 4-4: Signal fading. 44
Figure 4-5: The hidden and the exposed node. 45
Figure 4-6: Vulnerability period ALOHA & S–ALOHA. 49
Figure 4-7: Throughput in ALOHA & S–ALOHA. 49
Figure 4-8: Non–persistent CSMA. 51
Figure 4-9: p–persistent CSMA. 51
Figure 4-10 EY-NPMA. 52
Figure 4-11: Collision with the RTS / CTS mechanism. 54
Figure 4-12: The DCF - four-way handshake. 57
Figure 4-13: The DCF - the two-way handshake. 58
Figure 5-1: the SWAN model architecture [66]. 66
Figure 5-2: general behavior of a congestion controlled system [67]. 67
Figure 5-3: Architecture of the INSIGNIA framework. 69
Figure 5-4: INSIGNIA IP option. 70
Figure 5-5: 802.11e MAC layer architecture [72]. 73
Figure 5-6: EDCA access categories. 74
Figure 5-7: Interfame spacing / IFS in IEEE 802.11e [72]. 75

Figure 5-8: The topology information learned by A as a member of the core by beaconing. 79
Figure 5-9: Discovery split and Tickets distribution. 81
Figure 6-1: The layered architecture. 86
Figure 6-2: Notifications: the ECN. 90
Figure 6-3: additional interactions between layers for cross layer design. 91
Figure 6-4: Cross–layer information sharing. 92
Figure 6-5: LLE–TCP, the ARQ agent position [86]. 94
Figure 6-6: LLE–TCP in multi–hop networks. 94
Figure 6-7: MobileMan reference architecture. 95
Figure 6-8: Rate adaptive MAC + Minimum hop routing. 97
Figure 7-1: VoIP transmission. 101
Figure 7-2: Framing. 106
Figure 7-3: Jitter control. 107
Figure 7-4: E2E delay impact on Voice quality [94]. 109
Figure 7-5: MAC real time scheduler. 117
Figure 7-6: EFORTS design. 118
Figure 7-7: EFORTS State Collector. 119
Figure 7-8: Periodic state collection. 120
Figure 7-9: Network delay of VoIP traffic in a linear topology / G.711 – 802.11 (1997). 126
Figure 7-10: Network delay of VoIP traffic in a linear topology / GSM 6.10 – 802.11 (1997). 126
Figure 7-11: Network delay of VoIP traffic in a linear topology / G.723.1 – 802.11 (1997). 127
Figure 7-12: E2E delay of one VoIP session with different Codecs – 802.11. 128
Figure 7-13: the ORiNOCO 802.11b PC Card. 129
Figure 7-14: Network delay of VoIP traffic in a linear topology using G.711 – 802.11b. 130
Figure 7-15: Network delay of a VoIP session with GSM 6.10 and G.723.1 – 802.11b. 130
Figure 7-16: Network delay of one VoIP session with different Codecs – 802.11b. 131
Figure 7-17: bursts of losses and network delay of a VoIP session. 132
Figure 7-18: thirty nodes placed in a 650 x 250 m area. 133
Figure 7-19: Nodes movement. 134
Figure 7-20: E2E delay vs. time of VoIP flow from node 4 to 15. 135
Figure 7-21: E2E delay vs. time of VoIP flow from node 15 to 4. 136

Figure 7-22: E2E delay vs. time of VoIP flow from node 11 to 22.....137
Figure 7-23: E2E delay vs. time of VoIP flow from node 22 to 11.....137
Figure 7-24: E2E delay time of VoIP flow from node 22 to 11, 802.11b vs EFORTS.....138

LIST OF TABLES

Table 7-1:Speech coders.....102
Table 7-2: Summary of Attributes for 3 Commonly Used VoIP Coders [95]. 104
Table 7-3: Temporal parameters in conversational speech (average for Eng.) [96]..... 104
Table 7-4: Speech transmission quality [99]. 110
Table 7-5: The State collector – the head description. 120
Table 7-6: The State collector – intermediate nodes entries. 120
Table 7-7: Simulation parameters of the 802.11 MAC/PHY. 123
Table 7-8: Simulation parameters of the 802.11b MAC/PHY..... 124
Table 7-9: Selected Codecs and their attributes. 125
Table 7-10: ORiNOCO11b PC card spec [114].....129
Table 7-11: 802.11b vs. EFORTS deadline–aware MAC. 139

The beginning is the most important part of the work.
Plato

Chapter 1

1. Introduction and Preview

1.1. Motivation and challenges

An ad hoc network is a fully distributed network of nodes with radio interfaces. The nodes are allowed to move arbitrarily and share the wireless channel. They asynchronously access that channel to send data generally over multiple hops. This makes them very flexible networks which can be installed with the lowest costs and in the most critical environments. This pushed people to envisage many applications for theses networks.

Ad hoc networks are more required when other networks are missing. This is the case of battlefield, massive failure of existing infrastructures or isolated regions. However, they are also very interesting as a supplementary networking and communication mean in urban areas and voice communication is without doubt the most aspired service. Today, the number of mobile telephony users has far surpassed the number of internet users and this is strong evidence on the value of voice communications in the view of today's users. These facts have made of ad hoc mobile networks the subject of enormous research works in this last decade,

first, to offer usual internet services on these networks and second, to add new multimedia applications support to them.

However the most notable characteristics of an ad hoc network are a lack of resources. Using a shared wireless channel and limited power supplies add serious challenges to the designer of ad hoc networks. This apply for usual applications but when referring to multimedia applications such as VoIP, which still is not well supported on the Internet, the task becomes more challenging. VoIP applications require bounded delays and losses and continuous service. However, ad hoc networks are dynamic and unpredictable and the transmission delay in such networks is boosted by the multi–hop routing and the contention on the access to the transmission medium. The interferences also cause the frames to be retransmitted and this adds extra delay and wastes the network resources. The loss ratio is also important in wireless networks and this can cause communication to stop if a determined threshold is exceeded. To all that we can add the security issues which results from the use of the shared wireless channel as a transmission medium. To meet these challenges a new design approach must be investigated which must take in account the characteristics of these networks.

1.2. Thesis outline

In this thesis the problem of QoS support for multimedia applications is studied. We focused on VoIP as a new popular and emerging application. We resorted to a new design approach to optimize VoIP support in ad hoc networks through a more efficient use of the available resources.

The chapter 2 presents the environment on which our work will take part, namely, ad hoc networks and their position comparing to other technologies. Then we present a deep and critical study of two important layers which have a big impact on real time traffic support (MAC layer in chapter 3 and routing in chapter 4) in the context of ad hoc mobile networks.

We next studied the problem of QoS support in MANETs (chapter 5) and presented a representative set from the existing proposals in this direction from the literature, this with giving our views on the presented approaches.

The chapter 6 presents the cross layer design approach and taxonomy of the forms it can take. Some examples are given to show the importance of the approach and to justify our

choice. Also a cautionary snapshot is taken to show the limits and reverse effects that can results from such an approach.

In the next chapter (7), we focused on VoIP as the application on which our work will take part. We then described EFORTS, a cross layer design for optimizing VoIP support in MANETs. The simulation analysis and results that support our solution are given in the same chapter. The last chapter is the conclusion of our research work and also gives outline for future work.

If you can't explain it simply, you don't understand it well enough.
Albert Einstein

Those that know, do. Those that understand, teach.
Aristotle

Chapter 2

2. An Introduction to Ad Hoc Mobile Networks

Abstract

This chapter introduces the main concepts regarding Ad Hoc mobile networks. We have deeply examined their features figuring out the new offered opportunities by these networks as well as the new faced challenges.

2.1. Introduction

Wireless communications has known immense evolution in the three last decades. Today, their applications have been vulgarized and touch more and more our everyday life. The most beneficiary factor was probably the mobility, these networks offers more mobility surface over time. From first forms which were perhaps the cordless phones that can be used freely within a small region of some meters radius to actual cellular phones that permit to

users to move in a wide covered area, mobility has been continually improved at each new step. The advancements realized in this field are principally the answer to the progressive pressures made on by the evolution of our information society which is itself driven by the progressive technological evolution and penetration. Now, multiple mobile computing devices had penetrated our daily life from cellular phones, personal assistants to powerful laptops.

The concept of wireless communications is not new. Early applications can be traced back to the DARPA Packet Radio Network project in 1972 which has been inspired by the success of the packet switching technology [01] in the wired internets. The aimed goal was to interconnect mobile nodes in environments where there is no fixed base like in the battlefield. This is what we call an ad hoc mobile network. An ad hoc mobile network (MANET / Mobile Ad hoc NETwork) should permit to all its nodes to communicate trough other intermediate nodes whenever a physical link is available and without relying on any fixed infrastructures. The nodes are capable of move so that the topology is dynamic and arbitrary changing without loosing connection between every nodes pair. This technology enables networking in situations where we can't (Battlefield, massif damage of infrastructures by disasters…) or we don't want (costly choice) to exploit wired or cellular infrastructure.

Among all wireless networks classes, ad hoc mobile networks class seems to be the one which is not clearly present in our life. This can be considered as shocking since that it was the first wireless technology on which researchers start to study in the early 70's. But in the same time this also can be considered as natural if taking in account this class of networks characteristics and its original goals. In these last years, people start to think some commercial applications to this technology and who know, in few years, this networks class can enter our life from the wide door!

2.2. Wired vs. Wireless networking

In contrast to wired networks, wireless communications use a shared medium which is the radio channel. This means that the nodes can't use separate communication lines to connect with their neighbors so that they must take care to avoid that concurrent wireless transmissions corrupt each other. The radio channel can't be bounded and once the radio signal is launch, we can't estimate the exact distance it will reach in the space which make it

unpredictable and impose additional care about other properties such as signal transmission strength. Also, this channel can't be isolated or protected against foreign perturbations which cause more transmission errors and less reliability. Another major difference with wired communications is that in wireless communications a node is unable to detect collisions whenever it occurs while transmitting. Radio channel is also more exposed to eavesdropping, jam and spoofing from intruders.

All these limitations make the wireless links really unreliable but in spite of that, wireless networking stays a very interesting choice in many situations because it offers an incomparable flexibility and mobility comparing to wired networks. It is also useful in reducing networking costs in many situations since the installation of a wireless network requires considerably less cabling or no cabling at all. It is also suitable to use a wireless network rather than a wired one in temporary installations. Without missing that the deployment of a wireless network take very less time compared to the deployment of wired ones.

2.3. Wireless evolution

The wireless communications has known an exponential growth in the past decade. We have been witnesses of the great advances in network infrastructures, wireless applications and the vulgarization of the usage of wireless devices such as cell phones, PDAs and laptops, all getting smaller and more powerful in their capabilities and also have access to more applications and network services.

Radio transmissions can be considered as the first step on the wireless networking track. Next comes cordless phones which came to answer the need to communicate while on the move, or away from a fixed phone outlet and the MTS (Mobile Telephony System) which has been deployed in some cities in the USA in 1946. MTS, Improved MTS or Advanced MTS in Japan are analog systems which utilize a BS (Base Station) with a high power transmitter to cover the expected area and some dispersed receivers to rely mobile units to the BS. This system has many limitations due mainly to inefficient exploitation of the radio channel (limited simultaneous calls, frequent interferences…). These limitations find the answer in the first generation of cellular systems through the use of the cells concept, it consist of replacing the huge BS with a number of low coverage stations forming adjacent cells. The available

spectrum is partitioned in a manner that interdict two adjacent BS from using the same frequency bands to avoid interferences but permit their reutilization to non neighbor BS. The BSs are connected by a wired network and some mechanisms are deployed to allow mobility across cells. First generation cellular systems have recognized a big success which has surprised one and all regardless of the many disadvantages it comprises due mainly to its analog signaling nature. The second generation of cellular systems, first deployed in the early 1990's, were based on digital communications. Digitalization has offered additional capabilities such as encryption, errors correction, data transmission in highest rates used in text messaging and more efficient spectrum usage by means of some RF carrier sharing techniques. Second generation systems with many and incompatible standards offered a high voice quality and reliable service, but a low data rate transmission which make it impractical in some cases like for multimedia applications. 3rd generation systems come mainly to fulfill eventual future needs in the field by offering some improvements like highest data rate and more flexible architecture to smooth the interaction with other systems (The Internet for example). Even with the variety of applications offered by 3G, the basic services existing in 2G stay always the most used. Now, we speak about the future coming 4G systems. The major expected improvements concern the unification and the integration with other global services by using a common platform like an IP based core.

In fact, all this was only one side of the story, many other wireless networks has appeared and realized large evolution in the last decades like satellite communications (voice telephony, broadcasting, GPS, Internet access…), WLL (Wireless Local Loop) systems exploited in telephony wireless access to the public network and Wireless Data Systems.

Wireless data systems are dedicated to transmit pure data, they are packet switched networks and they allow connecting computers or other mobile devices. It is the case for WLANs (Wireless LAN) or PANs (Personal Area Network) which has a very short range peer to peer communication and permit connecting of mobile devices like mobile phones in a small radius area.

The different wireless systems listed above rely on some fixed infrastructure to perform networking. There is a different approach in viewing wireless networking which may get wide applications in the future. For example, if we need to perform some networking tasks on a set of mobile units where there is no fixed communication infrastructure, if every

mobile unit act as a router and try to relay packets to the other nodes, so we'll get a self–organized network. This is what we call and ad hoc mobile network. These networks are only formed when needed, and they are capable of networking without relying on any fixed infrastructure or centralized control. This networks class is expected to occupy a key role in future 4G networks!

2.4. Ad Hoc Mobile Networks

The ad hoc mobile networks (MANET) as can be understood from their name are temporary networks which are formed just for a specific purpose. They are infrastructureless networks, autonomous and entirely distributed systems where different wireless mobile nodes are capable of arbitrary movement and power switching without breaking the network connectivity since it is physically possible. In ad hoc mobile networks nodes are not required to reach every other node directly because of their limited range but they may rely on other intermediate nodes for relaying. Therefore they must operate as hosts and in the same time as routers. The use of such networks is suited when it is impossible or undesirable to use fixed infrastructures.

The concept of ad hoc mobile networks is not new unlike the naming, early ad hoc networking forms can be dated to the DARPA Packet Radio Network (PRNet) project in 1972 which was not a fully distributed architecture because of some centralized controls and which was inspired by the Internet and the packet switching technology. Another similar project was launched in 1983 to address main issues in PRNet, in the areas of network scalability, security, processing capability and energy management [01]. The Survival Adaptive Radio Network (SURAN) was based on link state routing contrarily to PRNet which used Distance Vector based routing. These two projects were purely for military purposes. The US defense aimed behind that to develop a technology which permits connecting different nodes in battlefield.

After that, interest on MANET has gone down until 1990's where it has been renewed by the new hopes that offered the reached technological evolution. The increasing evolution recorded in the segments of wireless communication and computing has made MANET more feasible than before. Therefore, this field has known increasing growth until today and draws more and more attention of researchers from academia, industry and military.

2.5. Characteristics of ad hoc mobile networks

In contrast of all other existing networks, Mobile ad hoc networks are expected to operate in hard and least equipped environments. Nodes are required to achieve networking operations autonomously and without being assisted by any fixed infrastructure. This eliminates any centralized control or administration. The mobility made possible by the wireless nature of the links between nodes shape the network topology and make it dynamic and unpredictable but the topology variations must not affect the nodes connectivity. Therefore many mechanisms must be deployed.

Mobile ad hoc networks inherit their characteristics from both distributed systems and wireless communications but add other ones due to their ad hoc nature:

2.5.1. Characteristics due to distribution

Mobile ad hoc networks are fully distributed systems. The forming nodes are all equivalent and no one run any singular network control task, so that the presence of any node is not vital for the network. Therefore the control tasks are distributed similarly on the comprising nodes. The distribution aspect makes MANETs flexible networks but harder to manage and control. We mean by flexibility here the vivacity of the network and its ability to tolerate eventual changes in the network composition and to fit the new resulting situations as softly as possible.

Distributed control allow too the autonomy of the network, we mean by that the ability to build networking without relying on any centralized entity. However this comes at a price, distribution of the control on all nodes adds other design problems. In such distributed operations and dynamic topology networks, designing and deploying an efficient solution for the network management becomes a challenging task.

2.5.2. Characteristics due to wireless links nature

The MANETs are wireless networks, so they inherit wireless communications problems as well as their advantages.

2.5.2.1. Limited bandwidth and highest loss ratio

The utilization of the radio channel as a support of communication reduces significantly the offered bandwidth. First, because the regulations on usage of frequency bands by the authorities are very rigorous and you are not free to use the frequency range you need because it is a shared resource which needs to be normalized. Also because the radio channel is a shared medium and can't be bounded, so in best cases the channel must be shared between all neighboring nodes to avoid interferences. As well, the loss ratio in wireless communication is very significant because it is impossible to protect the medium from outside signals and interferences are more frequent with both system and outside signals. The signal quality is exposed to degradation by noise too. The collisions are very probable because the transmitting node doesn't have the ability to listen the medium while transmitting with ordinary facilities because the strength of the transmitted signal and the fading property which make received signal weaker than the originated one. In addition to that, the wireless link is time varying because of the regular changes in the surrounding environment.

2.5.2.2. Security weakness

Wireless systems are more exposed to security threats because the radio channel can't be protected or driven in secure track, the radio signal propagation properties make it exposed to eavesdropping, jam and spoofing from intruders.

2.5.2.3. Mobility enabling

Using wireless links as a support of communication allow the key feature of ad hoc mobile networks. It is of course mobility. Mobility has really revolutionized our life style and fit very well to what human in the 21st century needs. Today, we spend more and more time on moving, at the same time, time becomes more precious and has really became money. Mobility allows you keep working whenever you are on your desktop or on the move.

2.5.3. Mobility

In mobile ad hoc networks there is no limitation or restriction on how and when any node will move. All nodes are free to move in the manner and moment it wants to do it. The only thing we may possibly know is the mobility model followed by these nodes. This in fact can be ended by studying their behavior and not by forcing any restriction. This offers a big

flexibility and convenience. But the impact of this on networking is very important because it makes the topology dynamic and unpredictable. Mobility creates big challenges for designers on several levels from the radio channel access control to connecting nodes and ensuring reliable services.

2.5.4. Multi hopping

In ad hoc networks the transmission range of the nodes is small and can't cover the entire topology, this in fact is one of the mains design choices of this class of networks and probably the most distinct difference between mobile ad hoc networks and other wireless communication systems. It offers a best usage and reuse of the available and precious bandwidth in both the spatial and temporal spaces and reduce considerably interferences probability. However the network nodes must count on the other nodes to communicate with non neighboring ones starting by one neighbor and hop by hop until reaching the destination.

2.5.5. Limited energy source

In accordance to their nature, mobile ad hoc networks must rely on portable power supplies which are limited sources. Therefore it is required to limit the overhead carried by each node. The energy efficiency problem makes the ad hoc networks more challenging. Finding efficient and less voracious solutions for this network's class is a big challenge today.

MANETs with these characteristics may be considered as an exceptional family among existing networks and we can guess that it will find many applications in the few coming days and in many domains but in the same time they reside more complex and constrained than any other existing network.

2.6. Ad hoc mobile networks applications

Ad hoc mobile networks seems to be completely absent from our everyday life. This in fact is principally due to the goal for they were originally designed. Historically, MANETs have primarily been designed to be used in particular circumstances where it may be impossible to rely on fixed bases to perform networking operations especially in battlefield or in massive infrastructures failure like in disasters recovery. The MANETs concept is very

original but hard to materialize. Delivering reliable services on such platforms reside a real challenge for researchers.

Today, a new wave has come. Numerous factors speak in favor of ad hoc mobile networking like technological progress and the generalization and wide usage of mobile devices. This with the improvements achieved these last years on MANETs has drawn the attention of professionals from several fields to the importance of this class of networks in our future life.

Future applications of ad hoc networks can only be limited by imagination. This is because the flexibility and mobility they offer and which fit very well to what we need in today's life. But in the same time, we must recall that big work is yet expected on MANETS to give them to a truly practical level.

If we try to enumerate the possible applications of MANETs today or in the close future, we must first start by tracks where they are the only candidates. I mean environments where no other networking form is possible because of the lack or massive failure of infrastructures. This is the case for battlefield, disasters recovery or isolated areas (Algerian Sahara for example). Another case of valuable usage of ad hoc mobile networks is what is called opportunistic or spontaneous networks, the goal is to connect roaming peoples in campus environments, conferences or anywhere people are gathering together to get better collaborative computing tasks and communications. In this case we can also envisage extending fixed networks and expanding their services to larger areas like offering Internet services to all nodes by connecting one or a set of nodes to the closest Internet access point.

2.7. Conclusion

Despite their promising features, ad hoc networks stay until now far from penetrating wide use commercial applications. Many ad hoc networking problems should be solved to offer acceptable services even with the big progresses achieved by the past. Big challenges are ahead but one thing is certain, after few years ad hoc networks will make it and enter our information society from the wide door. In the next chapter we discuss a key element in ad hoc networking. It is the problem of routing in those networks.

Study the past if you would define the future.
Confucius

Chapter 3

3. Unicast Routing in Ad Hoc Wireless Networks

Abstract

Routing in MANETs is a challenging task. In this last decade, it has drawn the attention of many researchers, many protocols have been proposed but only a few of them has leaded the track. In this chapter we tried to present a selected set of the main achieved works in this area.

3.1. Introduction

An ad hoc mobile network is a temporary network which is formed just for a specific purpose. It does not rely on any infrastructure or centralized administration which makes it autonomous and an entirely distributed system. The forming nodes are capable of arbitrary movement and power switching without breaking the network connectivity since it is physically possible through other intermediate nodes. This is because in ad hoc mobile networks nodes have limited transmission ranges by design and they are not required to reach every other node directly. That is why the nodes (or a part of them) must operate as hosts and

in the same time as routers. In addition to that the transmission medium is not reliable like wired networks and the effective bandwidth is limited by the medium characteristics.

These characteristics make of routing in MANETs fundamentally different from routing in classical networks. And the solutions envisaged for classical networks can't fit to MANETs. For example, wired networks use one or a small number of routers to connect a set of separated networks, but in the ad hoc wireless networks case any node can operate as a router. Thereby, it will have a variable and probably a big number of neighboring routers which produce a redundancy and significant additional overhead.

3.2. Routing in conventional networks

Today's routing protocols in conventional networks like the Internet are based on one of two primary algorithms, the distance vector algorithm or the link state algorithm. Distance Vector (DV) algorithm – also called Bellman–Ford algorithm because it is based on the Bellman equation and is a distributed version of the Ford–Fulkerson shortest path algorithm which is based on the previous relation too – periodically broadcasts routing information which consist of a vector of the learned distances from the entire network hosts but only to neighboring routers (where it will be used to infer shortest routes and update the local routing information). Therefore, the size of routing information to transmit become more and more important as the network raise in size and may slow down the performance of such algorithms. Also, because the DV algorithms are derived from the Bellman–Fulkerson shortest path algorithm, their convergence is very slow and any change in the network topology may require a considerable time to propagate and take effect on all the network hosts. First implementations of the DV algorithms was the RIP [10] protocol which has been widely used (routed for UNIX system, implementation from the Berkeley university) before being specified by the IETF in 1988 (the specification RFC in [10]). In 1994, the second version RIPv2 of the protocol has been published in [11] but the protocol kept important limitations like the formation of routing loops which cause the counting to infinity problem and impose a limitation on the maximal number of hops. IGRP & EIGRP from Cisco can be considered the best DV protocols (EIGRP is not a pure DV protocol) because they offer efficient techniques to cop with the known problems of the DV algorithms family.

In contrast, in Link state algorithm routers periodically flood the states of all the links with their neighbors in the whole network. This allows each router to get the global topology of the network and thus allows it to compute the shortest paths to all the destinations using an algorithm like the Dijkstra's shortest path algorithm. The routing information to exchange in link state based protocols is of small size but it must be flooded in the entire network and not only to the neighboring routers. There are two known implementations based on the link state algorithm which are actually used in classic networks, OSPF [13] and IS_IS [14] which are very similar. OSPFv2 has been published in [13] and is actually widely used in the Internet where it is very suitable for big and complex domains. OSPF is a direct application of the link state algorithm but it offers some supplementary techniques which permit to reduce the routing overhead like the partitioning of big domains into smallest areas.

3.3. Routing in ad hoc mobile networks

With the increasing advances achieved in the field of IT and wireless communications and the proliferation of numerous and more and more powerful mobile devices the ad hoc mobile networks have been focused by many researchers and are considered now as a promising networks class. But this class is at the same time full of challenges and its materialization opposes us to many design concerns. The design of efficient routing solutions in one of the key problems of MANETs and is truly a challenging task. It is probably the most popular among the other concerns in ad hoc networks. Therefore, the Internet Engineering Task Force (IETF) has founded the MANET working group. The initial aimed goals behind that was to standardize one or more Unicast routing protocols, and related network layer support technology which provides for effective operation over a wide range of mobile networking contexts, supports traditional IP service and also reacts efficiently to topological changes while maintaining effective routing in a mobile networking context [12].

The natural approach we may adopt in design of routing solutions in MANETs is probably the adaptation of the existing solutions in classic networks to the MANET case. However the existing solutions for routing in classic networks can't fit to MANET because of the fundamental differences between these two classes' characteristics and operations. In ad hoc mobile networks the wireless links are not symmetric obligatorily and a node may be unable to reach any node it is able to hear. Also, the topologies changes which are expected in

classic networks are completely of different nature from them in MANET case. In MANET the changes that can occur are not limited to the loss or reestablishment of links to some networks but the networks' nodes are not fixed and are constantly moving in an arbitrary manner. In addition to that, the mobile nodes must generally operate as routers which create a big redundancy. Therefore, big communication and computing overhead is produced which imply important energy and bandwidth resources consumption despite the fact that these two ones are precious resources in mobile ad hoc networks environments.

In fact, this can't invalidate the assertion that by some adaptations of classical routing solutions it will be possible to overcome some problems but at the same time some other problems stay very persistent and may require rethinking new solutions which fit well to the MANET nature.

The designed routing protocols should fulfill a few aspects to be appropriate to ad hoc mobile networks. The MANET workgroup of the IETF has listed some points to be carried [12]. A routing protocol for ad hoc mobile networks must provide loop free routes, according to the context it must utilize either on demand or proactive operation to establish needed routes. It must offer some mechanisms to cop with the common vulnerabilities of wireless networks. In addition to that, it must optimize to the maximum its operations to preserve the limited resources in both bandwidth and energy. We can add to that, the support of unidirectional links since it is a real fact in the MANETs context.

Numerous routing protocols have been proposed for ad hoc mobile networks. It is not in the scope of our work to make an exhaustive presentation on the achieved works in this area. However, routing is a key element in our work therefore we selected a set of routing protocols which we judged as representative and relevant to our purpose.

3.4. Classifications of routing protocols

There have been proposed many classification of routing protocols in ad hoc networks. In this section we gave the main used criterions.

3.4.1. Proactive VS On–Demand routing protocols

This classification is made on the activity mode adopted in the routes discovery. Proactive protocols try to maintain consistent routes between any pair of nodes and at all time by discovering and refreshing routes continually. They are characterized by smallest latency but also by a big overhead. On the other hand, On–demand routing protocols also called reactive protocols, try to establish routes only when needed and maintains them as long as they are in use. Therefore, they have a small overhead comparing to the previous class but more important latency. There is no comparison to do here because the importance of any approach is widely related to the context where it applies. It is apparent that proactive approaches are more suitable in situations where the network traffic is homogeneously distributed on the comprising pairs of nodes. However, reactive approach is more suitable when the network traffic is bursty and directed mostly toward a small subset of nodes. These two approaches have also been combined in some propositions (hybrid approaches).

3.4.2. Distributed VS source routing

In source routing approaches the source node must own the complete route to the desired destination. The route to follow should be transmitted with data so that the intermediate nodes can find the route to the destination in contrast to the distributed routing case where it is realized hop by hop until reaching the destination.

3.4.3. Flat VS Hierarchical routing

In flat routing the nodes are equivalent and run the same operations. In the opposite, with hierarchical routing the network is viewed as a set of clusters which are formed to offer better network's size scalability. Some nodes will have added tasks like leading or controlling a cluster.

3.4.4. Geographical approaches

These approaches have the particularity of using geographical information to help improving ordinary approaches. Also called location–based routing approaches, they need additional equipments to acquire the needed information like using a GPS.

There are other classifications in the literature but citing them here is not relevant to our purpose. We will present in the next sections the main existing routing protocols for MANET but we will ignore for the moment the geographical (location–based) approaches and focus on the ordinary ones.

3.5. Proactive routing protocols

Proactive routing protocols are constantly active in routes discovery and maintaining operations. Therefore, the routes are always available between any pair of nodes whatever they will be used or not. This allows avoiding important delays when sending data but it produces an overhead which may become significant as the size of the network raise in a critical environment such as MANETs where the resources are limited and precious. Such routing approach is neither new nor specific to ad hoc mobile networks since all the routing protocols in the Internet follow this same approach.

3.5.1. Destination–sequenced distance vector protocol (DSDV)

DSDV [15] is one of the earliest proposed routing protocols for ad hoc mobile networks. Therefore it is no more than an adjustment of an existing routing approach for wired networks. It is a distance vector protocol which is very similar to RIP [10]. The goal behind its design was to take advantage of the simplicity of distance vector protocols such as RIP but in the same time to add some enhancements to cop with the common problems of DV protocols family which consist essentially of the routing loops and the important produced overhead.

The introduced enhancement can be summarized in tow main axes:

3.5.1.1. Avoiding routing loops

The routing loops are common problem of DV routing protocols. This is due mainly to the routes learning mechanism adopted in these protocols which make it possible for a host to learn routes from his neighbors even if it was the source from they indirectly learned these same routes. With more simplicity, the inability of hosts to distinguish between stale and fresh routing information make it possible for a host to learn some routes which may comprise loops. DSDV adds an additional tag to each entry of the routing table; it consists of

an increasing sequence number which must be originated by the destinations and will be included in the exchanged routing information. Thereby, stale routes are differentiated from fresh ones. When detecting a link failure, nodes assign an infinite distance to all the routes which pass through the lost hop. The changed routes will then be advertised but they can't be tagged with the old sequence numbers of the lost destinations. Naturally, the sequence numbers of the routes toward any destination node should be originated only by this same node; however as seen in the preceding case; it may be possible that a node may have to originate infinite distance routes when it loses its links. To resolve this problem DSDV permits to use odd sequence numbers to tag an infinite distance route to a destination by other nodes. However, when a node advertises itself it uses even sequence numbers.

3.5.1.2. Reducing the control overhead

In DV routing protocols, the routing information are broadcasted periodically to the neighboring nodes. On receiving an update message the nodes make use of it in picking optimal routes. Since the routing information to broadcast concerns the entire topology, the message's size has to be certainly important. Therefore, their transmission and processing may result in a significant communication and processing overhead in both the transmitting and the receiving nodes. To lighten this load, DSDV utilizes two kinds of updates. One will carry all the available routing information called a "full dump" the other type will carry only information changed since the last full dump called an "incremental" [15]. The incremental updates will be sent every time a significant change occurs. The designers state that this significant change is the recognition of a new metric or a new sequence number. In an attempt to evaluate the performances of DSDV, an implementation has been realized in [16]; the authors have used two approaches in defining incremental updates. The first called simply DSDV triggers these updates only when a metric change occurs and the second called DSDV–SQ triggers them if either a metric change or a new sequence number is detected. They found that while DSDV–SQ is much more expensive in terms of overhead, it provides a much better packet delivery ratio in most cases [16]. In full dump the entire routing table is advertised where each entry contains: the destination identifier, the next hop to reach it, the sequence number and the distance. Because of big size of such updates they are transmitted relatively infrequently to reduce the bandwidth consumption. However, the period of these updates can be determined according to the mobility degree of the nodes.

Because of the utilization of the sequence number tags, the chosen optimal routes can change not only if a better metric has been received but also if a new sequence number has been received. By the adopted routes selection mechanism in DSDV, a new route is favored on an old when even if it has the worst metric. This can frequently happened because of the time skews between the different nodes which are due to the lack of synchronization between nodes. Such thing may drive a node to advertise some routes before receiving better ones just after so it must retransmit the new route once more. These fluctuations affect extremely the performance of the protocol. To cop with this problem, DSDV advertise infinite distance routes without any delay but delay the other metric or sequence numbers changes advertisement by an estimated delay called the settling time. It is the average time to get all the updates concerning a destination.

3.5.2. Optimized link state routing protocol (OLSR)

OLSR [21, 22] is a proactive link state routing protocol. As mentioned by its name OLSR add some optimizations to classic link state algorithm. The connection pattern in ad hoc mobile networks is completely different from it in classic wired networks. In ad hoc networks each node is supposed to act like a router therefore the number of links each node has is relatively high if we consider the size of the network. This big number of links makes it unfeasible to use the classic link state algorithm. OLSR proposes to reduce this load by selecting a minimal and effective subset of the links by each node and reduce the insignificant redundancy when flooding its links state information.

The OLSR node choose a set of its 1 hop neighbors in a manner that it can reach all its 2 hops neighbors through this selected set. The chosen nodes are called the multipoint relays of the node (MPRs). In figure 3.1 we can see node A which selected B, C, D as its MPRs.

This technique allows each node to reach any other node only by using the selected MPRs. This helps to control flooding in the entire network and to achieve it with lowest costs. In addition, the selection of MPRs permits to reduce the network links density which is very significant for a link state protocol.

In link state routing protocols the host periodically floods their links state information in the entire network. As well OLSR proceed with the same way however in OLSR the flooded information only concerns a subset of the node links. Indeed, the OLSR

node only diffuses the links states of its selectors namely the nodes which have selected it among their MPRs.

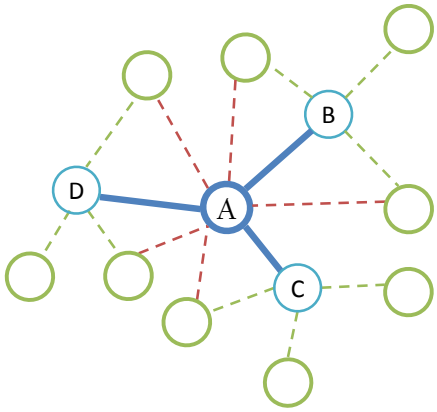


Figure 3-1: MPRs selection.

OLSR achieves the main operations of a classic link state routing protocol. It establishes neighboring relations with its 1 hop neighbors so that it can permanently know the state of its links. Also, it floods its links states information in the entire network. Finally and once receiving the links states from all the other nodes it computes the needed routes. OLSR adds to these three operations the selection of the MPRs.

The neighboring relation: the OLSR nodes establish a permanent neighboring relation with all the 1 hop neighbors. Thereby, the links states are known at each time. This is achieved by exchanging hello packets between neighbors. Each hello packet contains the list of bidirectional neighbors (the link symmetry has been detected) and the list of unidirectional neighbors (the link symmetry has not been observed yet) which can be promoted into a bidirectional one if the heard node receives it in a hello packet. With more details, a hello packet contains:

- A sequence number which is incremented every time the MPRs are modified.
- The list of all symmetric and asymmetric links.
- The state of the listed links which takes one of three possible values, bidirectional, unidirectional or MPR. An MPR is obligatorily a bidirectional link.

3.5.2.1. The MPRs selection

On receiving the hello packets from all neighbors, a node can learn all its 1 hop and 2 hops bidirectional neighbors. Selecting MPRs is achieved by selecting a minimal subset of the 1 hop neighbors such as this node can reach all its 2 hops neighbors through the selected subset (MPRs). Once the MPRs are selected they will be declared in the hello packets so that each node will be aware of the nodes that have selected it. The selectors are stored in a table named the MPR selectors table which is maintained by the hello packets.

3.5.2.2. Flooding the links states information

OLSR is a link state protocol therefore the nodes periodically declare their links states. But OLSR node only declare a subset of their links, indeed they only floods their selectors links states. Thereby an important redundancy is avoided. These updates are named the topology control messages (TC) and each entry is tagged by a sequence number which is originated by the selector itself and first broadcasted to the selected nodes in the hello packets. On receiving these messages the nodes updates their topology table. The sequence numbers help to distinguish old information from fresh ones.

3.5.2.3. The routes computing

When a node receives the TC messages from all nodes of the network it will get a sufficient topology map and can easily calculate any route. In link states protocols the most use technique is the Dijkstra's shortest path algorithm. OLSR suggests tracking back the source starting from the route destination.

3.5.3. Topology dissemination based on reverse–path forwarding (TBRPF) [25]

Topology Dissemination Based on Reverse–Path Forwarding (TBRPF) is a proactive, link state based routing protocol for mobile ad hoc networks. Like in OLSR the nodes periodically send beacon messages to maintain neighboring relations between nodes. Hello packets are used to this end. One Hello packet does not only contain the sender information but also the list of neighbors like in OLSR. In OLSR the list of neighbors is useful to select MPRs where TBRPF use it for an analogous task that we'll describe next. Each node running TBRPF computes a source tree to all the destinations which is formed by the shortest paths

to all nodes using the topology information it got. The key idea in TBRPF is to minimize overhead of link state approaches by only forwarding a part the link state information which is assessed to be useful. When a node receives an update from a neighbor it does not forward it immediately to the next hop as is the case in the standard link state approach. This information is driven to its target but in another form. Indeed, each node in the network will use the global information it owns to calculate an optimal source tree to all node and only a subtree denoted the reported tree is to be forwarded to a specific neighbor. Thus, we can summarize the operation of TBRPF in three functions, first the neighboring establishment, Second the topology propagation and discovery and finally the routes computation.

3.5.3.1. Neighbors discovery

The TBRPF Neighbor Discovery protocol allows nodes to detect their neighbor nodes and links breaks. It is similar to the OLSR neighboring establishment module. So Hello packets are used to detect links and their natures. Hello packets include the list of learned neighbors which allow the nodes to learn their 2–hops neighbors. This information is required to the topology discovery protocol. A new feature in this protocol is that it uses "differential" HELLO messages which report only changes in the status of links. This reduces the control overhead generated by the Hello packets. The protocol put information about the discovered neighbors in the Neighbors table where each entry contains the node ID and the state of the link (1–WAY, 2–WAY or LOST) and probably a custom metric. This module is independent and can be easily used with other protocols.

3.5.3.2. The topology discovery

Each node running TBRPF maintains a source tree which provides shortest paths to all reachable nodes. Figure 3.2-b shows the source tree maintained by the node S from the original topology shown in Figure 3.2-a. Each node computes and updates its source tree in function of the locally available information. A key contribution of TBRPF is that it allow to minimize overhead produced by link state protocols. This is done by only reporting a subtree of its tree to a specific neighbor. Of course for each neighbor a different subtree is reported according to its position. Reporting subtrees is done in periodic topology updates, and changes are sent in periodic differential updates of a smaller period.

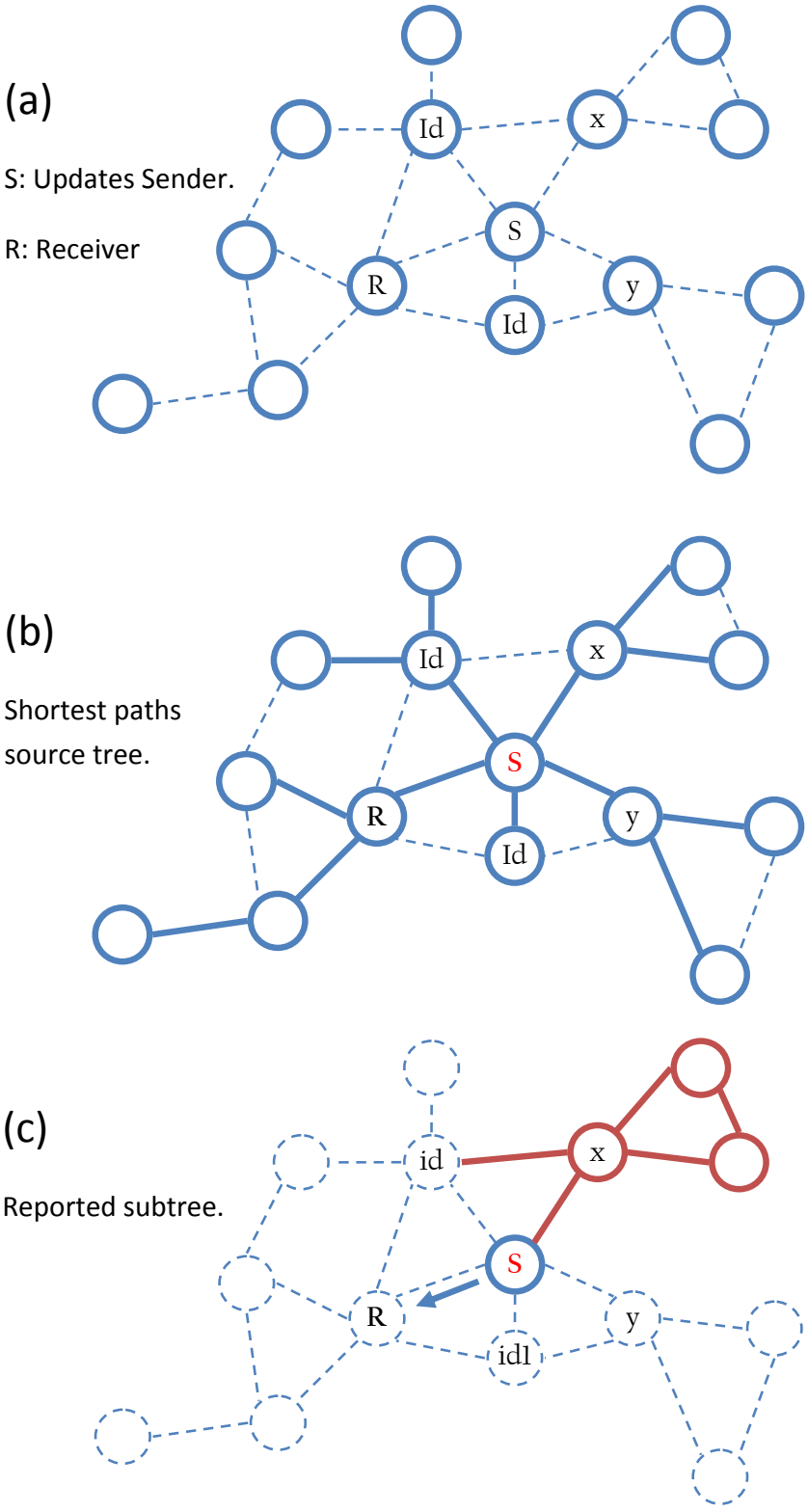


Figure 3-2: Topology information propagation in TBRPF.

A received topology update is implicitly forwarded since it can result in a change to the reported subtree. The reported subtree consists of bidirectional links such as one of the links ends is in the "reported node set" denoted ND.

Once a source tree is computed we can obtain a number of subtrees where each is routed in a neighbor. To form ND a subset of neighbors is selected and the nodes comprising the related subtrees will form the needed ND. The selection of these neighbors is achieved as depicted in figure 3.2-c. the reported subtree and the ND are shown in bold outlines. First the node to which is the target of the update is excluded (S). Also all the nodes which can reach this target in a shorter path than the 2–hop path via the sending node are also excluded (id). This simply means that neighbors which are one hop from the target of the updates are excluded since the distance of the path via the sending node is always two hops to any other neighbor. Nodes which are two hops from the destination also can be excluded. This is because ties are broken using relay priority or routes Ids.

3.6. Reactive routing protocols

Reactive routing protocols adopt an innovative approach in achieving the routing task. Called On–demand routing protocols too, they computes routes only when needed. The main characteristic of these protocols is that they produce relatively small control overhead. This is because the nodes don't waste efforts on establishing routes which have not been requested yet. However this may induce some delays in transmitting data. Such protocols have been designed to give an alternative to the weighty proactive protocols and which may not be tolerable in a critical environment such as ad hoc mobile networks. Generally, reactive protocols achieve three major functions: creating, maintaining and deleting routes which are events driven.

3.6.1. Temporally Ordered Routing algorithm (TORA)

TORA [17] is an on–demand distributed routing protocol which belongs to the family of algorithms referred to as "link reversal algorithms". It is partially based on the Gafni–Bertsekas algorithms [18] but it does not share their inconveniences related to the reaction to graph partitions formation. The key feature in this protocol is that it permits to limit the propagation region of the topology changes to the places where they occur. "It decouples the

generation of potentially far–reaching control message propagation from the rate of topological changes [17]". In addition to that, TORA offers loop–free and multiple routes to any destination.

TORA operations are comparable to two techniques presented in [18] and which allow transforming a DAG (Directed Acyclic Graph – see fig. 3.3) into a destination–oriented one. A destination–oriented graph is a graph where all the comprising nodes have an oriented path toward a specified destination. These two techniques are the full reversal technique and the partial reversal technique. In full reversal technique each node which does not has any outgoing link reverses the direction of its incoming links. The algorithm will converge if graph is connected. In partial reversal method if a node does not have any outgoing links it chooses one of its incoming links which haven't been reversed and inverse it. If its entire links have been reversed it reverse them all. TORA uses a modified version of partial reverse technique. Whenever a link failure at a node causes the node to lose its outgoing links to reach the destination a series of link reversals starting at that node can bring the DAG back to a destination–oriented state. TORA allows fast detecting partitions and eliminating them.

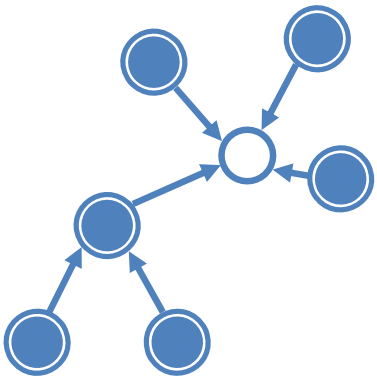


Figure 3-3: Destination oriented DAG.

TORA operation can be summarized in three functions: creating routes, maintaining & erasing routes. Before describing these three operations it is indispensable to present some notations which help us describing the protocol.

3.6.1.1. Model

- The network is modeled as a graph $G(N, L)$, where N is a finite set of nodes and L the set of existing links which are initially undirected.
- Two nodes are neighbors if and only if a link exist from *i* to *j*, from *j* to *i* or completely undirected.
- A vector value is associated with each node where these values can be totally ordered and represent the heights of the nodes.
- A link between two nodes *i* and *j* is directed from *i* to *j* only if *i* is higher than *j*.
- A node is said to be a local minimum if it is lower than all its neighbors and local maximum if it is higher than all its neighbors.
- The height of the node *i* is given by the quintuple $H_i(\tau_i, oid_i, r_i, \delta_i, i)$. The first three values represent a reference level and the last two ones represent a delta with respect to this reference level.

τ_i: *time of the reference level definition.*

oid : *the originator id.*

r : *a flag which indicates if a reference level is reflected.*

δ : *used to order nodes within the same refelected level.*

i : *the node id, it guarantees that the nodes which have the same reference level and also the same* δ *can be totally ordered.*

3.6.1.2. The protocol description

Initially the height of each node is set to NULL except the destination:

$H_i(-,-,-,-,-)$ And $H_{dest}(0,0,0,0,0)$.

In addition to its own height, every node maintains the list of the heights of its neighbors which are initially set to NULL. The nodes maintains a list of link states too which depend of their heights and the heights of their neighbors. A link can take three possible states, UP (upstream: the neighbor is higher) for incoming links, DN (downstream: this node is higher than its neighbor) for outgoing links and UN for undirected ones (the neighbor

height is NULL). When a node's height is NULL it is considered higher than any not NULL height.

a. Creating routes

The routes discovery is achieved using two sorts of packets, The Query and the Update packet (QRY and UPD). The QRY packet contains the required destination identifier while the UPD packet in addition to that the height of the node which broadcast it. When a node needs to establish a route with a destination node it broadcasts a QRY packet and set up a flag (the route required flag, RR).

On receiving a QRY packet:

- If the receiving node RR flag is set the QRY packet is ignored.
- If the receiving has no outgoing links (a local minimum) then it re–broadcast the QRY packet and set up its RR flag.
- If the receiving node has at least one outgoing link and its height is non NULL then it broadcasts an UPD packet unless it has been broadcasted before.

On receiving a UPD packet:

- If the receiving node RR flag is set then the node's height is updated as follow: it will be set to the minimum height among its neighbors then increments its δ : $H_i(\tau_j, oid_j, r_j, \delta_{j+1}, i)$ / j is the neighboring node which has the lowest height. Once it sets its height, the node updates its link states according to the new height and rebroadcasts an UPD packet.
- If the RR flag of the receiving node is not set, that node simply updates its link states.

In the all cases, the received heights must be recorded first. In figure 3.4-a the node D start a query to discover the destination. In figure 3.4-b the node Dest reply by the UPD packet. Figure 3.4-c show the propagation of the UPD packet in the network until the route is established between D and Dest in figure 3.4-d.

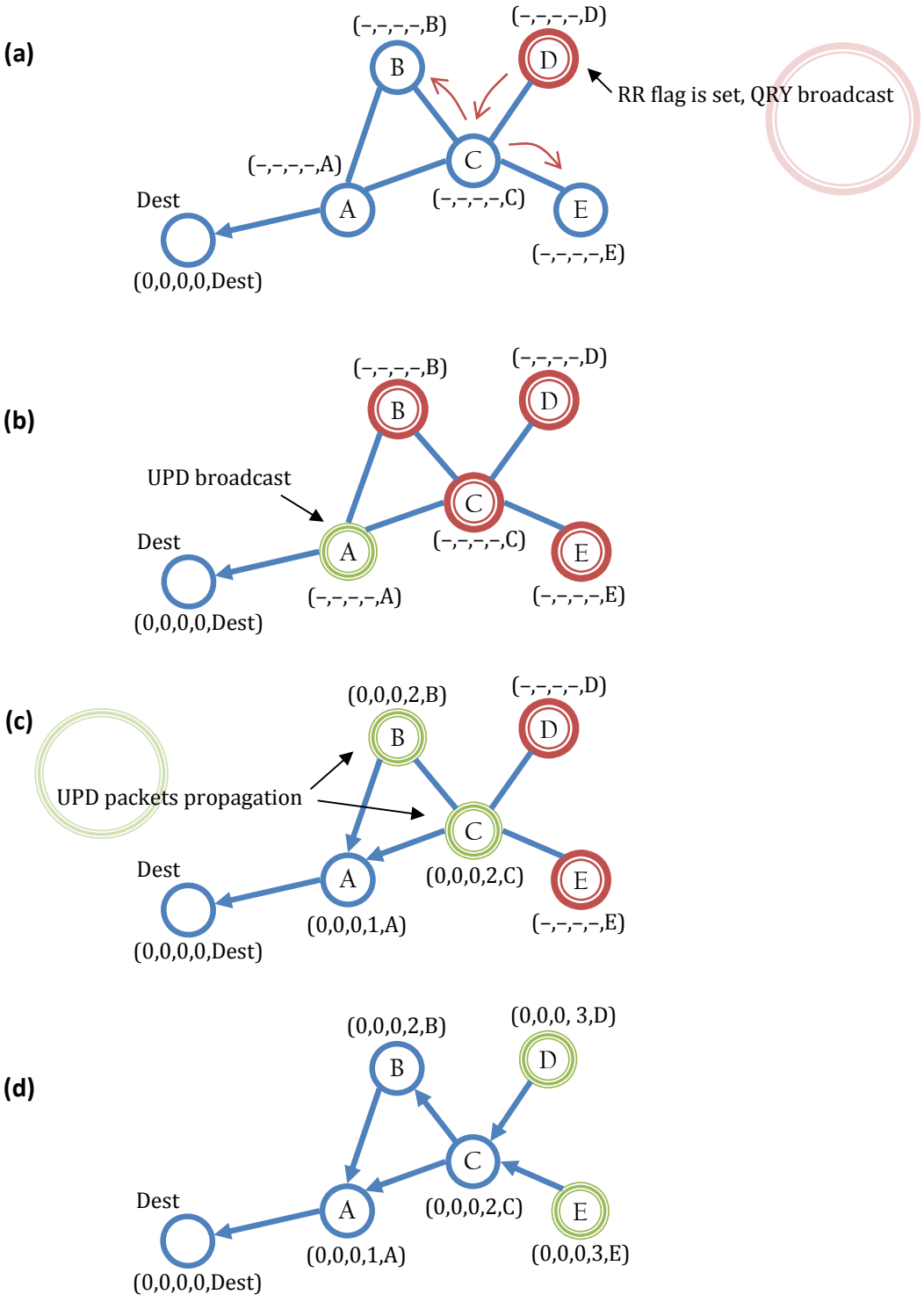


Figure 3-4: Routes creation in TORA.

b. Maintaining routes

Maintaining routes consists of the restoration of the lost routes to a destination. The reaction to such event is done by any node which has a non NULL height namely the nodes which belong to the formed destination–oriented DAG.

When a node which belongs to the formed DAG becomes a local maximum it launches the routes maintaining process. This can happened due to some links failures, nodes failures or links reversal caused by some updates. The totality of the operation can be recapitulated in five cases:

(1) The node has lost its last outgoing link because of a link failure: (fig. 3.5a). In this case and if the node has other incoming links it originates a new reference level and its height becomes: $H_i(t, i, 0,0, i)$ / t is the time of the occurrence of the failure. To simplify the description we suppose that the nodes have access to the same physical clock however in the implementation we can make use of logical clocks. In the case where the lost link was the last active link the node simply set its height to NULL.

(2) The node has lost its last outgoing link due to a link reversal following the reception of an update message: In this case, the node looks at the list of its neighbors' heights. If there is more than one reference level, the node takes the highest one and among all the nodes with this reference level this node adopt the lowest height by taking the smallest delta number and decrementing it by one: (fig. 3.5b)

$H_i(\max ref\ level, \min \delta_{\max ref} - 1, i)$

(3) The node has lost its last outgoing link due to a link reversal following the reception of an update message: In this case, the node looks at the list of its neighbors' heights. If there is only one reference level, and its third value which define the route reflection is not set then it adopt a higher reference level by setting *r=1*: $H_i(\tau, oid, 1,0, i)$. (fig. 3.5c)

(4) Similar to 3 but the route reflected flags are set to 1 and the node is itself the originator of the common reference level. In this case the new defined reference level has been reflected to the originator. The node deduces that there is no route to the destination and that a partition has been formed so it sets its height to NULL and initiates erasing invalid routes (fig. 3.5d).

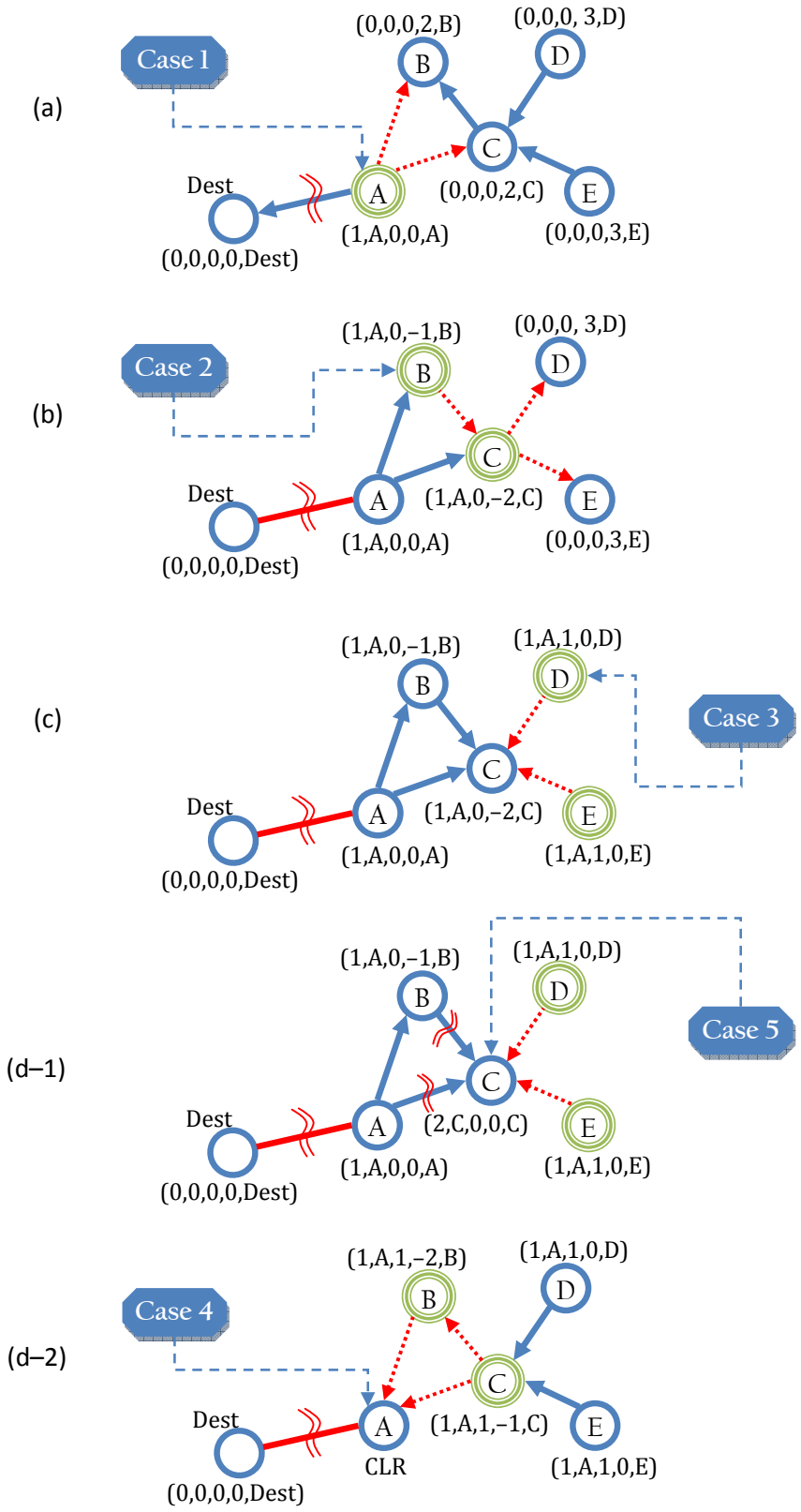


Figure 3-5: Maintaining routes in TORA.

(5) Similar to 4 with the difference that the node is not the originator of the reference level. In this case the node must have lost a link toward the originator. So that the node defines a new reference level by itself (fig. 3.5d).

c. Erasing routes

When a node detects a partition (fourth case in maintaining routes) it sets its height to NULL then initiates a routes erasing process by broadcasting the CLR (clear routes) packet. The CLR packet includes both the destination identifier and the reflected reference level. On receiving this packet, the nodes set their heights and the heights of their neighbors to NULL; thereby the routes of this partition will get deleted.

TORA provide an unusual approach to discover routes in ad hoc mobile networks. The main advantages of this protocol are that it allows multiple and loop–free routes (short term loops) to any destination, it also allows to detect the network partitions and limit the range of propagation of the topology changes to the region where they occur. In spite of its being efficient in finding multiple and valid routes, the optimality of these routes is not a primary concern of TORA; the computed routes becomes less optimal as the time passes.

3.6.2. Dynamic source routing

DSR [19, 20] is an On–demand source routing protocol, in source routing protocols the transmitting node is responsible of providing the complete route to the destination so that the intermediate nodes just follow it. Therefore the route is carried in the packets headers. Like all reactive protocols DSR avoids wasting additional efforts in maintaining routes which it may not need them soon. Once the route is learned, the node caches it in its routes cache and will continue to use it until it become invalid. While using any route, it is possible that this route get broken due to the mobility of the hosts. DSR utilize a specific mechanism to repair such situations.

Thus, DSR achieves two main functions, the routes discovery and their maintenance. The first allow nodes to learn the routes and the second help to keep them while they are in use.

3.6.2.1. The routes discovery

When a node desires to reach any destination which it doesn't know the route, it starts the route discovery process. It broadcast a route request packet which will be flooded in the entire network (fig. 3.6). This packet contains the targeted destination identifier, this node's identifier, a request identifier to control the flooding operation and a routes record the record which contain the list of the nodes to traverse to get to the destination. The routes record is updated at each hop by adding the identifier of the current node. Once the RREQ (route request packet) reaches its target the destination node replies by a route reply packet (RREP) to the originator of the request.

On receiving a route request for the first time, the node checks the target identifier in the packet. If it wasn't the target it simply rebroadcast the packet otherwise it replies the originator by a route reply packet which contains the needed route.

To send the route reply to the originator, the target node needs too to have the route to the initiator of the route discovery. If the network links are supposed to be symmetric the target may reverse back the route reply packet back on the same route. Otherwise, the target too must flood a route request packet to reach the original route discovery initiator but in this case the route reply will be held on (piggybacked). Once the route reply reaches the originator of the route discovery it caches it and uses it to route packets.

3.6.2.2. Advanced operations in routes discovery

In the previous section we have said that the targeted node is expected itself to reply the originator of the route discovery. However, it is possible to save additional efforts by using the cached routes to reply the route requests. When a node receives a route request packet and the sought route is available in its routes cache. We can think that this host replies it from the cache instead of the target. Thereby an important CPU and communication overhead which results from flooding the network by such packets is avoided. However additional problems are expected. Indeed, allowing such replies to other hosts than the destination means that we can get multiple and almost certainly simultaneous replies which induce a considerable redundancy and can drive into collision or congestion in the network. In this case some techniques can be deployed to cop whit these effects like delaying replies and discarding non loop–free routes. The responding node must also take care if the received

route request packet is piggybacking any data to the targeted node; in this case the node must create new packets to send this data to the destination with falsifying the transmitter identifier.

As we said before the RREQ are flooded in the network to look for their target. This means that this operation is costly and must be handled with care; indeed, in addition to using the request identifier which allows nodes to determine if a received request has been already sent, when a needed destination is not reachable the requesting node must deduce that and use appropriate techniques to avoid saturating the network by insignificant packets like using exponential delays.

Finally, it is possible to learn some routes for free by enabling promiscuous receive mode. This certainly induces additional CPU overhead but in the same time it can save important bandwidth which is the most critical in common ad hoc mobile networks.

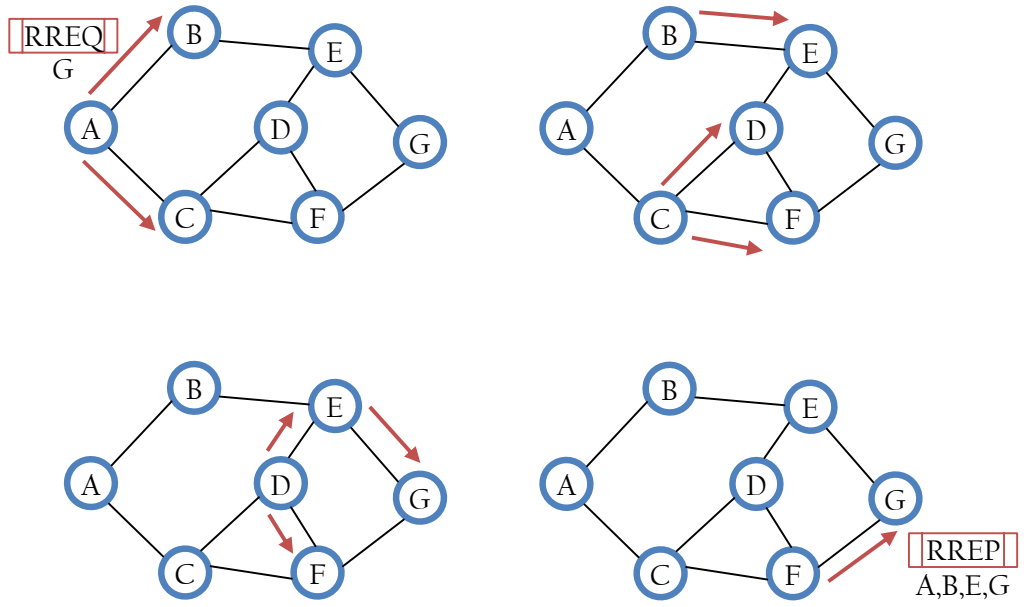


Figure 3-6: Routes discovery in DSR.

3.6.2.3. The routes maintenance

Routes discovery and routes maintaining in DSR are event driven. A route is computed only when it is needed and similarly it is only renovated when a failure occurs on any hop of the route. Therefore, the nodes may be able to detect their next hops losses

whatever they occur; this can be done through requiring acknowledgments of the sent data in some layer level. Some MAC layer protocols propose such hop–by–hop acknowledgments; indeed, hop–by–hop Acks can be achieved in both link and network layers. This makes it possible to detect the lost hops as quick as possible and also to localize them in contrast to end–to–end Acks, the unique possible mechanism which can be adopted in transport or application layer.

Using the hop–by–hop Acks the node which detects the failure is the node which has lost its next hop. If any node on the route is unable to deliver data correctly to the next hop then it must return a route error packet which comprises the lost hop to this data's originator. For that, this node must own a route toward it. If no route is available (this implies that the links are not symmetric) this node must initiate a route discovery to the originator and may be piggyback the route error packet on the route request packet. When the originator receives the route error packet, it checks its routes cache and truncates the routes entries which contain the lost hop and reinitiates the route discovery to deliver data.

In end–to–end Acks, the node can deduce the occurrence of a failure on the route if it can't receive an Ack within a limited delay. In such situation the node deletes completely the route from its cache and restarts the route discovery.

3.6.2.4. Advanced operations in the routes maintenance

The routes maintaining efficiency depends greatly of how soon the routes breakages are detected. Enabling promiscuous receive allows node to get free information about possible hops losses. Such information allows nodes to be aware of important changes in the routes and act more quickly to avoid undesirable delays. It can also aid in optimizing routes. For example, if a node is supposed to reach another node on the route in a few hops but it discovers trough promiscuous listening that this node has become a neighbor so it can shortcut this route and notify the originator about the change trough a route reply packet.

The routes errors must reach all the nodes which use these routes to get consistent routes every time. Because the links asymmetry is supported by DSR, the route to any destination may be different from the returning one. So when a hop is lost on the route then sending back the route error packet can take a different way and the nodes forming the route can stay not informed about this important change and keep stale routes. To solve this

problem we can envisage that the originator must retransmit the route error packet again toward the end node on receiving it. Figure 3.7 show a link departure between nodes E and G. Because this information is sent to the node A through a different path, node A just resend this error message.

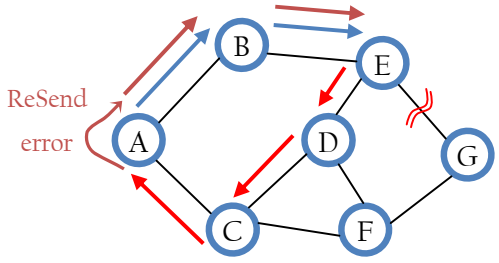
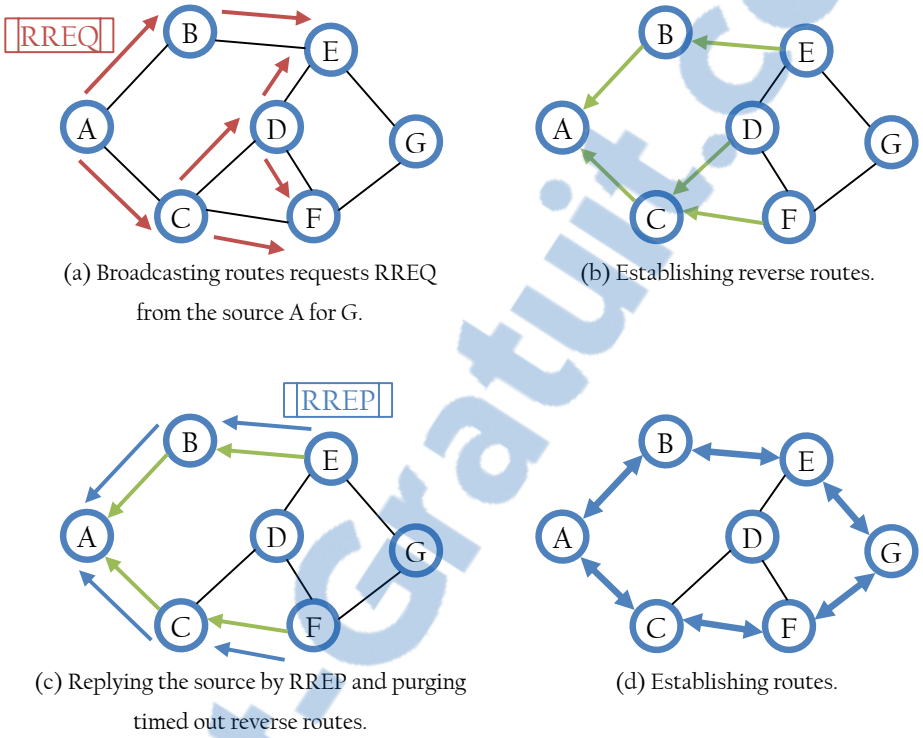


Figure 3-7: Resending route error packets by the originator DSR.

3.6.3. Ad hoc on–demand distance vector routing protocol

AODV [23, 24] is an On–Demand routing protocol which is similar to DSR but it uses distributed routing rather than the source routing adopted in DSR. Also and unlike DSR, AODV doesn't support asymmetric links. The routes discovery in AODV is achieved by flooding a RREQ in the network. The RREQ packet includes the both the destination identifier and the last known sequence number for this destination so that the replying nodes shouldn't propose a lowest SQ number for this destination. The RREQ packet also contains the request and the source identifiers which together determine a unique RREQ. In addition to that it contains the hops count which is initially set to zero and the source sequence number which can be useful in intermediate nodes once this one become requested by any node. Flooding such packets in the network creates multiple routes the source. These routes are established by reversing the routes taken by the RREQ packets and are associated with an expiration timer to purge the route if it is not adopted by the source node. Indeed, when a node receives a RREQ packet it sets a pointer the node from it receives it to get a hop–by–hop reversed route to the request source. The RREQ packets can be replied by the destination itself or by any intermediate node which has a fresh enough route to the destination. Therefore, the source node can receive multiple replies and must choose the freshest and the most optimal one. If no route is available then the receiving node of the request must

rebroadcast it if it was the first received request; this last is decided by both the request and the source identifiers.



(a) Broadcasting routes requests RREQ from the source A for G.

(b) Establishing reverse routes.

(c) Replying the source by RREP and purging timed out reverse routes.

(d) Establishing routes.

Figure 3-8: Routes creation in AODV.

The RREP packet is returned to the source using the reverse route created as the RREQ was forwarded. By the same manner, the route to the destination will be created as the RREP packet is returned back to the source. Once the route is created, it must be maintained as long as it is needed. A timer is associated with all routes in each node and the routes will be removed if not used for a sufficient time. A route that has been recently utilized is considered as an active route and its timer has to be refreshed. When any failure occurs along the route the source node must be notified so that it will reinitiate a new route discovery if it always needs this route. The node upstream of the break invalidates the routes to each of those destinations in its route table and sends back a route error (RERR) packet to its predecessor on the route. Each node on the route which receives the packet will in turn invalidate the broken route and send back the RERR packet until it reaches the source.

We have taken a look in TORA, AODV and DSR which comprise the main concepts used actually for on–demand routing. Other protocols have been proposed in the literature like

Dynamic MANET On–demand routing (DYMO) [34] which is a clone of AODV developed in by the IETF. It implements additional tuning to fit small and multi–interfaces devices.

3.7. Conclusion

Routing in ad hoc wireless networks is an active research board. It is a little early to judge the performance of the proposed protocols. The IETF has until now ended the specification of OLSR, DSR, AODV and TBRPF and other protocols are still under revision. The most simulation results are obtained under specific assumptions on the size and the dynamic of the network. For example reactive protocols are not efficient if the network dynamic is high and proactive protocols generate more overhead than needed in networks with low mobility. Therefore, it is not possible to get a protocol for all ad hoc networks. Deciding about the finest protocol to adopt depend on where it will apply. We think that it will be of importance to illuminate more the concept of MANETs and define the different possible application scene and then define protocols for each subclass according to its parameters. Routing really plays a central role in providing high networking performance for mobile ad hoc networks. However, also other stack components have a big impact on that performance. In the next section we present the MAC layer protocols design for ad hoc networks as a key factor in offering good performances in those networks.

Those who cannot remember the past are condemned to repeat it.
George Santayana

Chapter 4

4.MAC LAYER PROTOCOLS FOR AD HOC MOBILE NETWORKS

Abstract

MAC protocols design for ad hoc mode is very challenging and delicate. In this chapter we exposed the MAC layer design issues for ad hoc networks. We presented a selected set from the literature. The presented protocols involve the main concepts proposed in the literature.

4.1. Introduction

Ad hoc mobile networking mode is a new philosophy in the networking area. It promises high flexibility and freedom. However, there are many obstacles in the road of making it a practice. Ad hoc mobile networks use the radio channel to carry transmissions. This medium allows the required mobility but being a shared medium necessitates particular and well designed Medium Access Control (MAC) protocols to organize its utilization. The wireless communication are extremely unique, they are far different from point to point connections, and also different from multi point connections like Ethernet segments because

in wireless networks it may be impossible to hear some nodes that use the channel in contrast to the Ethernet case. Therefore, redesigning new MAC protocols for ad hoc mobile networks is indispensable. These protocols must consider the different features of ad hoc networks and the requirements of the applications.

4.2. The multiple access to wireless mediums

It is common in wireless networks that many users attempt to use the radio channel simultaneously. Efficient allocation of the available spectrum resources between users is a key design aspect of access protocols. When dedicated channels are allocated to users it is often called multiple access [09]. It is suitable for many applications which require continuous transmission and delay constraints. Contrary to that, bursty applications don't necessitate such way in allocating the transmission channel; it may be allocated only when needed. Bandwidth sharing using random channel allocation is called random multiple access or simply random access [09].

Multiple access techniques also called multiplexing techniques divide the available transmission channel into sub–channels and assign them to different users. The most common methods are time–division multiple access (TDMA), frequency–division multiple access (FDMA) and code–division multiple access (CDMA). Directional antennas add an additional option, they allow the division of the transmission space between users, this technique is called space–division multiple access (SDMA). Multiplexing allows sharing the radio channel capacity between the contending stations. Multiple transmissions are multiplexed to a common channel.

4.2.1. Time division multiple access

In TDMA the radio channel is divided into time slots where in each slot only one user is allowed to send (Fig. 4.1). Because of periodicity TDMA is more suitable for continuous traffic like voice communications.

TDMA offer a virtual channel to each user even if it dose not need continuous transmissions which wastes the available resources. This problem can be lightened by weighting the affectation of timeslots. A major difficulty of TDMA is the requirement for synchronization among the different users. This synchronization is achieved in common cases

by some centralized entities. TDMA is used in the GSM (Global System for Mobile communications) digital cellular phone standard.

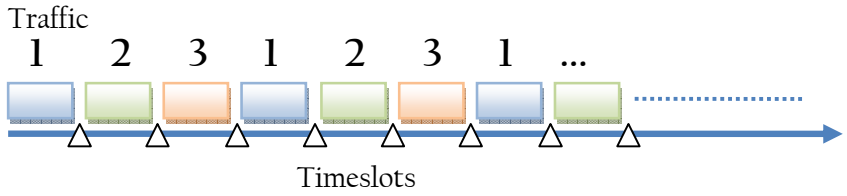


Figure 4-1: Time Division Multiple Access.

4.2.2. Frequency division multiplexing

FDMA splits the radio channel into a few number of non–overlapping frequency channels. Each user is assigned a frequency channel and can use it all the time which allow multiple nodes to transmit simultaneously (fig. 4.2). Of course like in TDMA, if a channel is unused the channel is effectively wasted.

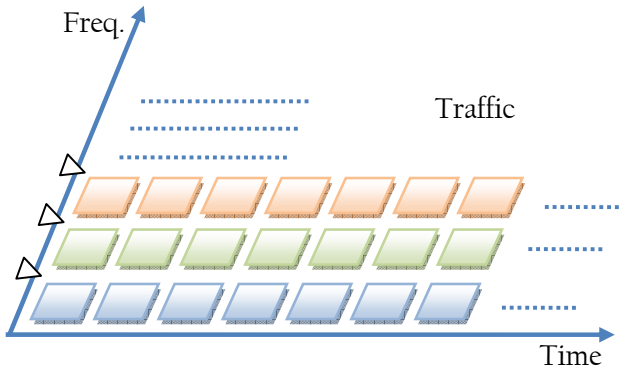


Figure 4-2: Frequency Division Multiple Access.

The partitioning of spectrum into frequency channels is achieved by modulating the corresponding carrier frequencies with the data to be transmitted. At the receiver, the distinction between other stations' signals is reached through filtering. Guard bands are required between channels in order to avoid adjacent channel interference between neighboring frequency bands. The separation of multiple users through frequency division multiplexing implies many guard bands and thus spectral inefficiency [44].

4.2.3. Code division multiplexing

In CDMA the signal to transmit is modulated by a spreading code. The resulting spread signal occupies a wider bandwidth. Simultaneous transmissions are allowed and the receiver uses the spreading code to separate out the different senders hence a different code is used by each user. For a receiver, the other users' signals appear like noise therefore the bigger the number of users the higher the noise floor and the number of stations transmitting in the same frequency channel is also limited.

4.2.4. Space division multiplexing

SDMA uses direction as a dimension in signal space. The same bandwidth can be used in the same time in separated spaces without interference (fig. 4.3). This is achieved with directional antennas. SDMA allows a better utilization of spectrum resources but needs further information on users' positions at real time. In mobile networks realizing SDMA is a tricky task.

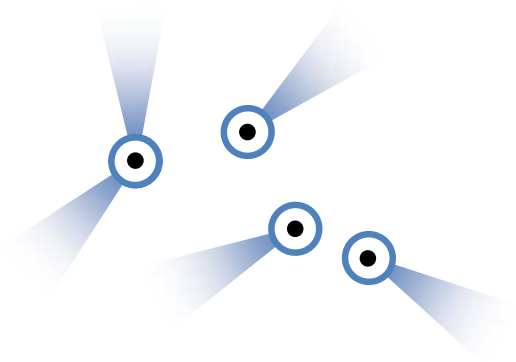


Figure 4-3: Space Division Multiple Access.

These techniques may be combined to make a hybrid division multiple access. Many systems today use a combination of multiple access schemes. The multiple access schemes seen above require a minimum knowledge on the users' distribution in the network to achieve such static divisions. They are suitable for continuous traffic like voice since dedicated channels are reserved even if they are not used. In ad hoc networks the topology is dynamic and unpredictable. Furthermore, all the nodes have the same rank and hold the same responsibilities so no centralized entity is available. In addition

to that, traffic may be bursty or continuous. These facts make the use of such scheduled access schemes in the ad hoc context unfeasible.

4.3. Ad hoc key considerations for Mac protocols design

The role of the Medium Access Control (MAC) protocol in ad hoc wireless networks is to control and coordinate the access to the shared channel. This coordination must be achieved in a distributed manner because of the lack of infrastructures. This protocol must allow the network resources to be efficiently exploited by the comprising nodes with the lowest possible costs. To reach this goal, many considerations are to be taken in view of the fact that the ad hoc mobile networks are unique environments.

4.3.1. A shared and multi–access medium

The radio channel is a shared medium and cannot be restricted. Due to the fading property of signals, a node has limited range and limited area it can reach. But this area can intersect with other node's coverage areas which make multiple and the simultaneous attempts to access to the same channel possible. When designing MAC protocols for this class of networks this point must be considered. Of course, multi–access is not a unique characteristic for ad hoc mobile networks. It is the case for all multi point networks. However, in the case of ad hoc networks two nodes' transmissions can collide even if these nodes can't hear each other which make the common used techniques non applicable like the CSMA/CD protocol used in Ethernet segments.

4.3.2. Varying channel condition

Mobility and varying environment alter directly the channel quality which can require dynamic transmission parameters. Unlike wired channels, wireless channels are time varying and location dependent. The MAC protocol may consider this fact to achieve optimal service by adapting some parameters like frames size, transmission power, error control techniques...

4.3.3. Errors ratio

The wireless medium is error prone. This is because the transmitted signals are more exposed to noise and interferences with the other nodes' transmissions. Therefore, the MAC protocol should use efficient techniques to cop with this problem. Retransmission seems to be the straight solution. By using an ACK for each transmitted frame, the frames whose acknowledgments are not received within a timeout are to be retransmitted. Other techniques have been suggested like reducing the frames size and forwarding correcting codes rather than the entire frames.

4.3.4. Signals properties (speak XOR hear)

In CSMA based protocols the node first senses the medium to check whether it is idle or busy when attempting to send. If it find that the medium is being used the node defers its own transmission to an ulterior time. Otherwise, the node begins to transmit its data. In CSMA/CD, which is use in Ethernet LANs, the nodes sense the medium even while transmitting which allow it to detect collision immediately. However, CSMA/CD cannot be adopted in ad hoc networks because the node signal can collide as well with non reachable nodes transmissions as cited above (multi–access). Furthermore, even if a node is capable to hear some other nodes when it is silent, it will become deaf to them if it is on transmitting. This is due to the fading of the signals. In wireless networks, unlike wired networks, electromagnetic signals are transmitted in free space where signal strength fades in proportion to the square of distance from the transmitter (see fig. 4.4) which makes the presence of a signal at the receiver unremarkable if the receiver is about sending some data.

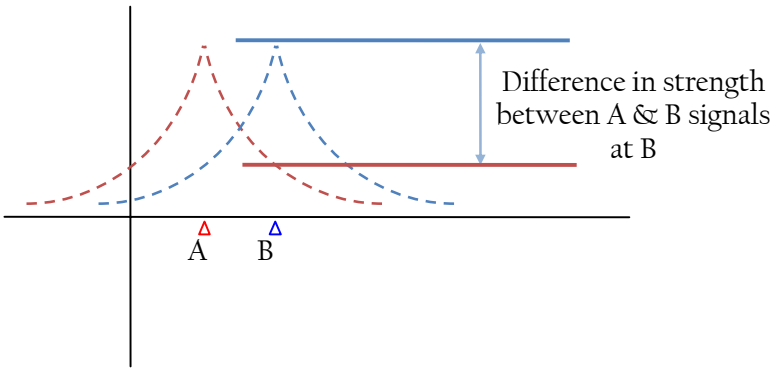


Figure 4-4: Signal fading.

4.3.5. Hidden terminal and exposed terminal problems

These two problems are well–known in ad hoc mobile networks and how the MAC protocol answers them decides vastly of its performance. They results from the way the access to the transmission medium is accomplished especially for the multi–access property of the medium and the distributed nature of these networks. The hidden terminal problem has been first mentioned in [40]. This problem occurs when a node transmits some data to a receiving node while this node position is invaded by the transmission of another node which cannot be heard by the first one (a hidden node). This situation results in loosing one transmission at least because of the collision at the receiver (see fig. 4.5).

The second problem occurs when a node desires to send some data but finds itself forced to defer its transmission to avoid collision because it sensed the medium busy with a communication it does not belong to. We say that this node is exposed (hence, the name exposed terminal) and has lost an opportunity to send its data (see fig. 4.5).

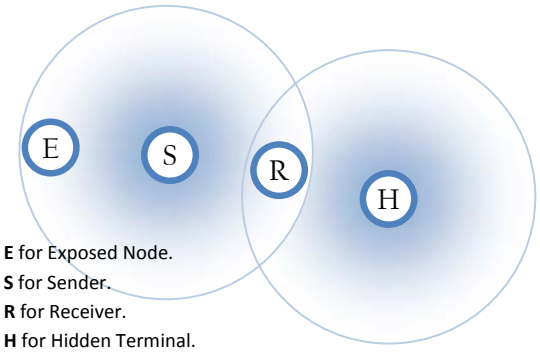


Figure 4-5: The hidden and the exposed node.

This was the most important issues to consider when designing a MAC protocol for ad hoc mobile networks. However, it is also important to consider additional issues like energy limitations, unidirectional links, QoS and security issues.

4.4. Design goals for MAC protocols in MANETs

A MAC protocol for ad hoc networks is supposed to allow multiple users to access the shared radio channel in a controlled manner. It allows users to share the finite available amount of radio spectrum. This task must be achieved while the best possible performance is

given. The main goals that a designed protocol should aim are optimal utilization of the network resources which are the radio channel and the energy. As well, it must aim to help supporting varied applications including real–time applications and to guarantee both stability and scalability. There are multiple metrics to decide about the performance of MAC protocol where each metric explicate one side of its rendering. The designer must consider all these metrics but may favor one or more among the rest. In this section we will to present the key aimed performance factors.

4.4.1. Controlled latency

The latency is the end–to–end delay from the moment a packet is queued at the source to the moment it is received properly at the destination. Usually, Latency is varying and constraints on it consist of fixing the maximum bound.

4.4.2. Throughput and goodput

The Throughput is the amount of data successfully transmitted from a source to a destination (bps) where the goodput is the amount of useful data successfully transmitted.

4.4.3. Fairness

It is the ability to access the medium equally by the contending nodes. This results in the fair sharing of the available bandwidth. Designing a fair MAC protocol is very difficult in the ad hoc context because of the divergence in the needs of each node and the varying number of contending neighbors to mobile nodes. The degree of unfairness is affected by many factors like the distance, the back–off mechanism adopted in many protocols and the location of the nodes in the topology.

4.4.4. Controlled overhead

The produced overhead must be of lightweight. It is the amount of auxiliary data and processing to supply to achieve the needed operation.

4.4.5. Power efficiency

Mobile nodes use carried battery power as source of energy. This source is limited and should be preserved as long as possible. Therefore, MAC protocols design should be achieved

in harmony with the limited power of the nodes. This can be achieved by reducing retransmissions and using just the needed power to deliver packets.

4.4.6. QoS delivery

With the convergence of data and voice networks and the proliferation of mobile devices, ad hoc networks are envisaged to support most of the networking forms known, even multimedia applications. Therefore, it is necessary to the designed MAC protocols to be aware of such applications and to offer feasible services at their level. This task is very difficult to accomplish because of the poorness of MANETs.

4.4.7. The physical layer facilities

Wireless communications has seen growing development. New advancements are expected from engineering in this field to allow more sophisticated services to the higher layer. MAC protocols designer should be conscious about the offered opportunities by the physical layer and exploit them.

4.5. Review of MAC protocols for ad hoc networks

As we have seen, the access to the medium in wireless networks can be achieved by means of scheduling approaches. It is the case for TDMA, FDMA or CDMA. Of course these schemes (called contention free schemes too or conflict free) don't fit to the ad hoc case as sensed before because ad hoc networks don't rely on any centralized administration and have a dynamic topology and an unpredictable shape. The users here would better use the entire bandwidth when they need to transmit their data. This approach is more suitable for ad hoc networks. The access in this case is contention based and users send bursts of data whenever they get present. As a consequence, multiple users might try to send data at the same time which can result in collision. If a collision occurs it is usually necessary to retransmit all the colliding data therefore contention based access can induce significant delays. Many works have been done by the researchers in the direction of designing appropriate MAC protocols for ad hoc mobile networks. In this section we will describe some important single channel MAC protocols for ad hoc networks.

4.5.1. ALOHA [45]

It is among the first proposed protocols for packet radio networks. According to this protocol a node transmits whenever the data to be sent is ready and without sensing the carrier. Because multiple nodes can simultaneously attempt to send in the same time, collisions are expected. In this case the damaged frames should be detected and retransmitted. Collisions can be distinguished by acknowledging received packets. Thus, a sender can always find out whether its frame was destroyed. Now if a node detect that the frames it sends has been destroyed it just waits a random amount of time and sends it again. The waiting time must be random or the same frames will collide over and over. The idea behind this protocol seems to be very simple and evident. But, can this protocol ensure good enough performance? Let assume that the frames transmission time is fixed and equal to τ and the frames arrival follow a Poisson law at a rate of λ frames per second. So, the channel utilization ratio $R = \lambda \times \tau$. R designates the number of frames to send per frame time. If ($R > 1$) the system is overloaded because the transmission rate exceeds the channel capacity. The Throughput T is given by this formula:

$T = R \times P$ *[no Collision]*

In addition to the new frames, the nodes also have retransmissions. Let assume that the probability of attempts per frame time, old and new combined, is also Poisson, with a rate of λ' frames per second. Of course, $\lambda' \geq \lambda$ but at low load $\lambda' \approx \lambda$ because of the small amount of retransmissions. The throughput becomes $\lambda' \times \tau$. To avoid collision with a transmitted frame, all the other neighbors should hold down their transmissions τ seconds before the transmission and until its end which make a break of 2τ.

The probability of n frames being generated in 2τ is given by:

$$P(n) = \frac{(2R')^n}{n!} e^{-2R'} \quad \Rightarrow \quad P(0) = e^{-2R'}$$

Therefore, overall throughput is:

$$T = R' e^{-2R'} = \lambda' \tau \times e^{-2\lambda' \tau}$$

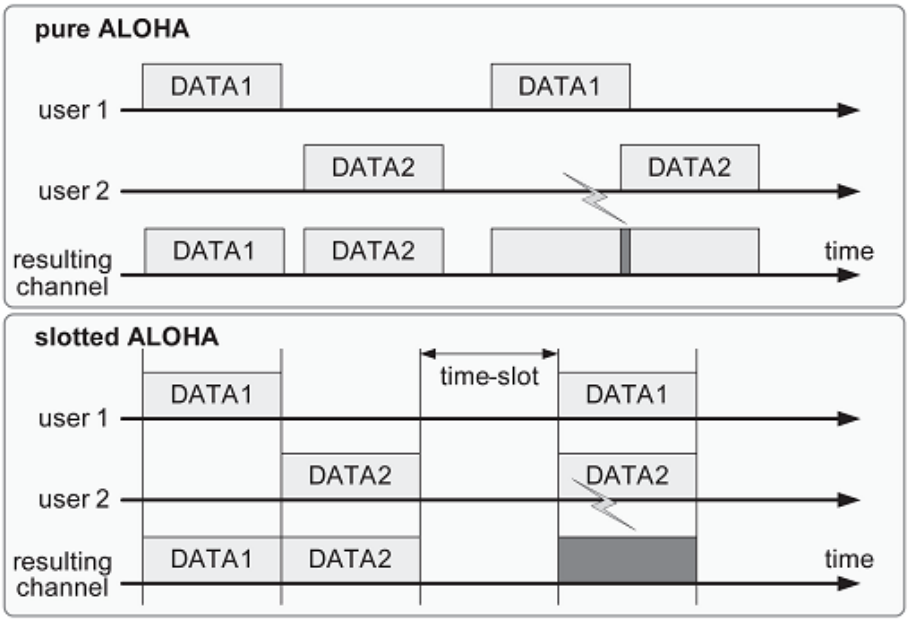


Figure 4-6: Vulnerability period ALOHA & S–ALOHA.

The maximum possible throughput of the system is *Max*(T) =1/2e=0.18 (18 %)at R'=0.5 which means that the maximum throughput is recorded for an arrival rate of one frame per two frames time. this throughput is limited to 18% of the total channel capacity which is mediocre (see fig. 4.7).

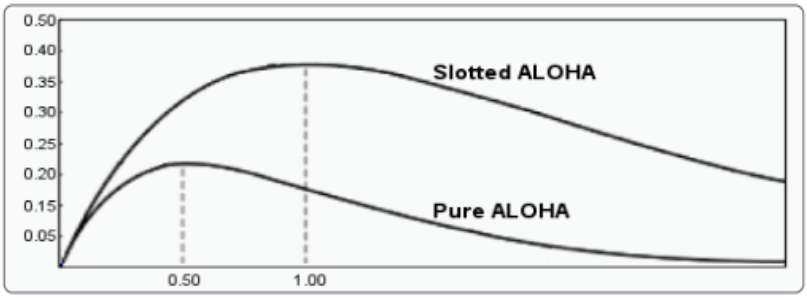


Figure 4-7: Throughput in ALOHA & S–ALOHA.

4.5.2. Slotted ALOHA [46]

It is a slotted version of ALOHA as confirmed by its name (also called S–ALOHA). The idea behind is to divide time into discreet intervals called slots and send frames whenever they arrive but only at the beginning of these slots. This technique reduces the vulnerability period to 1 slot (which is evidently the frame time) but requires synchronized clocks at all the transmitters.

The throughput becomes:

$$T = R' e^{-R'}$$

Thus, the maximum possible throughput of the system is doubled and recorded at R'=1 which means that the maximum throughput is recorded for an arrival rate of one frame per frames time (see fig. 4.7).

It is clear that both ALOHA and Slotted ALOHA can't be a put into application in ad hoc mobile networks because of their low performance and also the synchronization requirements for the latter. However, this does not disallow us to notice an important advantage which is simplicity.

4.5.3. Carrier Sense Multiple Access

The highest throughput that can be achieved using S–ALOHA system is 37% of the medium capacity. It is possible to improve the efficiency of ALOHA systems if we take in account the state of the medium before transmitting data. This can be done by sensing the medium to notice if either busy or clear. If the medium is sensed busy the transmission is rescheduled to a next random time. Thereby, evident collisions are avoided and more channel utilization efficiency is attended.

4.5.3.1. Non persistent CSMA

In this protocol the node which aim to transmit sense the medium first. If the channel is idle then the node transmits the frame otherwise it waits for a random time before retrying the same operation.

4.5.3.2. p–persistent CSMA

In p–persistent CSMA, the node senses the medium before transmitting. If the channel is busy, the medium stay waiting for it to be released and then transmit with a probability of p and defers the transmission with a probability of *1–p*.

4.5.3.3. 1–persistent CSMA

It is a p–persistent CSMA with p=1. So a node waits the medium to become idle and then send its frame.

It is clear that these CSMA variants will allow better performance comparing to ALOHA. However, CSMA is not sufficient to prevent collisions. CSMA doesn't consider the hidden terminal phenomena. Thus, collision may occur. Nevertheless, the main ideas of both CSMA and ALOHA forms the core of the majority of the proposed protocols for ad hoc mobile networks.

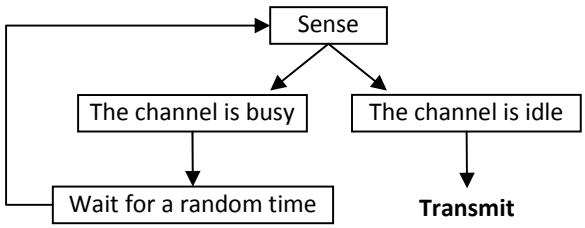


Figure 4-8: Non–persistent CSMA.

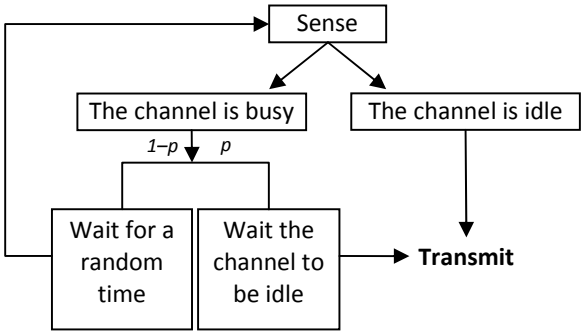


Figure 4-9: p–persistent CSMA.

4.5.4. Elimination yield non–preemptive priority multiple access (EY–NPMA) [47]

This protocol is proposed for the access procedure in HiperLAN1. High Performance Local Area Network type 1 (HiperLan 1) is a standard from the European Technical Standard Institute (ETSI). It supports multi–hop communications, multimedia applications and distributed networking but can also be adopted for centralized schemes since nodes can operate like gateways or access points. The operation of this protocol is divided into three

phases. The prioritization phase, the contention phase and the transmission phase. In the prioritization phase a priority is calculated taking in account the queuing time of the related frame. Once the access priority of the frame is decided the node senses the medium for a period which is function of the picked priority. If a priority of *a* is decided this duration is simply *a x timeslot*. After this duration and if no signal has been detected the node send a priority assertion burst. In the opposite, if a priority assertion of another node has been detected the current transmission cycle is abandoned. The contention phase starts after that between the nodes with the same priority. It is done in two phases. The elimination phase and the yield phase. In the elimination phase the node transmits for a random period of time and then listen the medium. If it finds it free then it passes to the next phase otherwise it is eliminated. Thus, the nodes that picked the longest period are victorious. In the yield phase the node sense the medium for a number of slots. This number is picked randomly (the distribution of the random numbers are such the odds of having only one winner are close to 100%). If during this period the channel was idle then the node starts its transmission. It is the transmission phase.

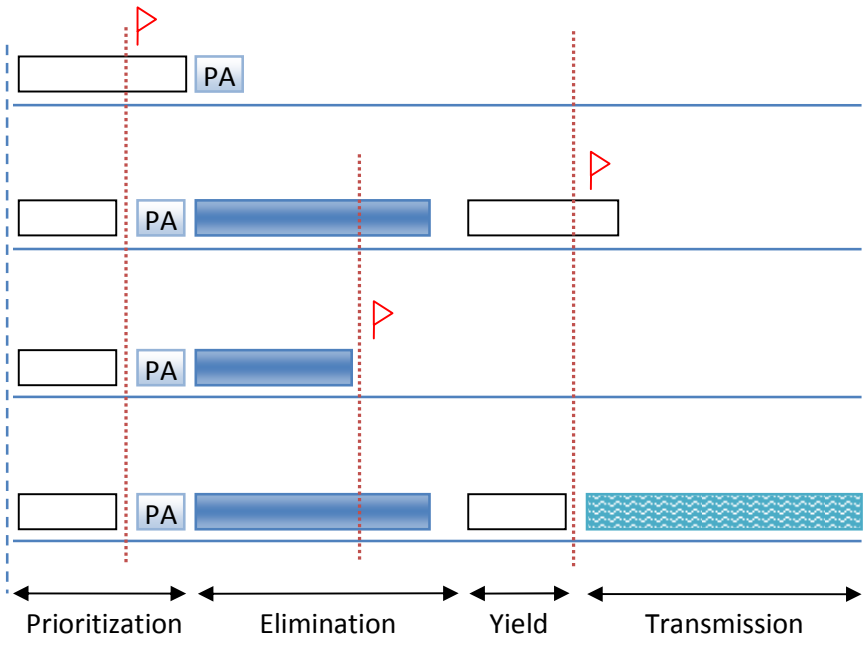


Figure 4-10 EY-NPMA.

EYNPMA tries to resolve collision that results from simultaneous transmissions. It achieves that by introducing qualifying phases before transmission where only one node should pass them. It is relevant to observe here that this protocol doesn't consider the hidden terminal problem. Thus, collisions may occur. As well, the overhead is pretty high (a node will got tired before passing to transmission). In addition to that it cannot carry hard QoS constraints even with its prioritization capabilities. This may explain why HiperLAN/1 has not met a big success in the market. The successor HiperLAN/2 is very different from the original. It adopts a centralized architecture which fits well to WLANs and not a bit in ad hoc networks. Therefore, we will not investigate it more.

4.5.5. Multiple Access Collision Avoidance (MACA) [49]

MACA has been proposed to overcome the hidden terminal and the exposed terminal problems. In MACA, before sending data a small packet should be sent to acquire the channel. This small packet is called RTS packet (Request To Send). After sending an RTS the node must wait for a reply from the receiver. Indeed, when the aimed destination receives an RTS it replies by a CTS packet (Clear To Send). Once received the CTS, the node can start its data transmission. This mechanism is efficient because when receiving an RTS the nodes which are in the range of the sender can deduce that their neighbor is attempting a transmission and accordingly wait for the CTS. The nodes which can't hear the CTS are free to send. Hence the exposed node problem is solved. In the other side the nodes which can hear the CTS must defer their transmissions to the end of the ongoing one. This after consulting the data transmission length included in both the RTS and the CTS packets. The CTS is also useful to avoid collisions with hidden nodes' transmissions because nodes which receive the CTS defer automatically their transmissions. Even if this mechanism reduces highly the risk of collisions, it doesn't completely eliminate the hidden terminal problem (illustration in fig. 4.11). Let's suppose four nodes A, B, C and D. A attempted first to send some data to B so it sent an RTS and received the related CTS. The CTS should be received by all the neighbors of B and especially those which are hidden to A. let's suppose too that C is in the range of B but out of the range of A (a hidden terminal). If the CTS transmitted by B collides with an RTS from another node D the node C will not be conscious of the transmission from A to B and subsequently can cause a collision at B if it attempts to initiate a transmission.

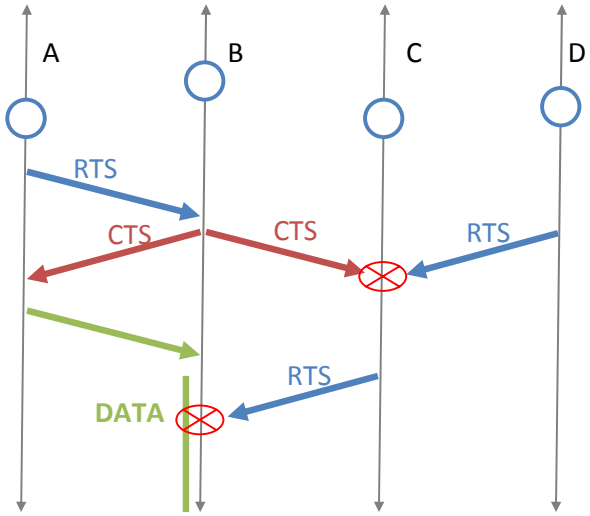


Figure 4-11: Collision with the RTS / CTS mechanism.

If a node that initiated a transmission by sending the RTS packets is unable to receive the related CTS it will eventually time out and assume that a collision has occurred. In this case it reschedules the packet for retransmission. MACA uses the binary exponential backoff (BEB) algorithm to select this retransmission time. This algorithm consists of selecting by a uniform distribution a random number between 1 and the BO (the back off counter). Every time the node fails to receive a reply to its RTS it doubles its BO. Once the CTS is received, BO will be set to BO_{min}.

4.5.6. MACA for Wireless (MACAW) [50]

MACAW has been proposed to overcome some problems in MACA which have been illustrated by simulation in [50]. The authors claim that the BEB algorithm generates unfairness in accessing the medium and the channel can be easily captured by a node. Therefore they propose another back off algorithm (Multiplicative Increase and Linear Decrease – MILD). Upon a collision, the backoff counter is multiplied by 1,5 and decreased by one unit at each success. This value is advertised in the control packets and the nodes copy this value whenever they receive it In addition to that MACAW suggests reliable transmissions by the use of acknowledgment packets. This help to reduce delays induced by the original protocol where the retransmission of the lost packets is carried by the higher layers (after timeout) and then waste big time. The whole steps of the protocol are as follow: RTS – CTS – DS – DATA – ACK. The DS (Data sending packet) is sent by the initiator to

mention that the RTS – CTS is successful since some exposed nodes may be unable to hear the CTS packet and hence may cause collision with the ACK packet.

4.5.7. MACA by invitation (MACA–BI) [51]

This protocol aims to reduce the delay of packets transmission in MACA. Indeed, in MACA a sender must receive the CTS before its starts the data transfer. The waiting time between RTS – CTS is considered as an additional delay. MACA–BI proposes to initiate the transmission by the receiver. When a node is ready to receive data from a neighbor it sends an RTR packet (Ready To Receive). On receiving this packet the sender begin the data transmission. Of course, in this case the receiver must be aware of the plans of its neighbors. This information can be piggybacked in the transmitted packets. Thereby, MACA–BI can operate efficiently with regular traffics but may degenerate with bursty traffic patterns.

4.5.8. Dual Busy Tone Multiple Access (DBTMA) [52]

DBTMA uses two separate channels for signaling and one channel for data transmission. For that reason it is usually classed in the literature as a multiple channel MAC protocol. In this chapter we will not present many proposed MAC protocols. We will just focus on basic ones and those relevant in the context of our work. Therefore we will not study multiple channel protocols. But we consider that DBTMA is a single channel protocol or a pseudo multiple channel protocol since it only use one data channel. DBTMA is build upon BTMA [53] (busy tone multiple access) which has been proposed for a centralized topology. In BTMA, the base station place a signal on the busy tone channel when it senses the channel to be busy to avert the other stations that access is currently denied to the channel. As the opposite of MACA variants where collisions are possible even between the control packets, BTMA use a second channel to protect the data transmission. Similarly, DBTMA use an out of band signaling to solve the hidden and the exposed terminal problems and hence avoid collisions and exploit the channel more efficiently. In DBTMA, the node which intends to send some data sends an RTS first on the data channel but sets up a transmit–busy tone on the first signaling channel. The receiver will not reply by a CTS but just sets up a receive–busy tone on the second signaling channel. Thereby, the nodes in the vicinity of both the sender and the receiver will be aware of what is happening. DBTMA solve completely the hidden terminal and the exposed terminal problems. The hidden nodes are averted by the

receive–busy tone all along the transmission and furthermore they can reply to an RTS and become receivers without harming the ongoing transmission of their neighbors. The exposed nodes can make the right decision according to if any receive–busy tone is sensed or not. In fact, DBTMA can achieve high efficiency but at the same time require more complex hardware. In addition to that it does not make use of acknowledgments.

4.5.9. Floor Acquisition Multiple Access (FAMA/FAMA–NCS [56, 57])

The protocol suggests some improvements on MACA. FAMA adds the carrier sensing procedure in both the physical and the virtual type to MACA. If the node detects an ongoing transmission or the handshake fails it will back off. The length of RTS packets is long enough to avoid the reception of the entire RTS packet while it hasn't reached some other nodes yet. In addition to that, the CTS packet is longer than the RTS. This allows a node which has sent an RTS to differ itself its transmission if it can sense the final part of any CTS. Therefore, this CTS packet is called a dominating CTS and acts like a busy tone to prevent hidden nodes from harming the signal at the receiver. The differing duration is one maximum packet in all cases. This drive to a bad space reuse especially when the RTS / CTS fails or if the data packet is small.

4.5.10. The distributed coordination function of the IEEE 802.11 (DCF) [59]

It adopts a CSMA with collision avoidance (CSMA/CA) protocol and involves concepts from both CSMA–based MAC protocols and MACA. Collision avoidance is the one allowed substitute for collision detection technique allowed in wired networks. It is achieved via the reservation of the channel before transmitting by means of some specific techniques. The IEEE 802.11 specifies two modes of operation. The PCF (point coordinated function) for centralized networks like WLANs where the standard has encountered a big success and the DCF for distributed networks which are the subject of our work. In the latter no base station is needed unlike for the former. The DCF protocol adopts two access mechanisms. One is a four–way exchange (fig. 4.12), RTS–CTS–DATA–ACK just like seen for MACAW and the other is a two–way handshake (fig. 4.13), i.e., DATA / ACK. In MACA variants the risks of having simultaneous transmissions' initializations are high because there is no coordination

between the nodes which share the radio channel. Sensing the medium before transmitting will absolutely involve lowest risks. However, it is always possible to have collision between RTS packets if they are sent almost in the same time. The idea behind the DCF of the IEEE 802.11 standard is to wait for the channel to be idle and the wait for a random time. If this time expires and the channel is always clear the handshake process is started. Otherwise, the node will differ to the end of the current transmission and restart the same procedure with the remaining time. In this way, the odds to get a collision between simultaneous RTS packets are reduced.

In more details, a node which aims to initiate a transmission waits the channel to be clear. Once the channel is idle, the node waits for an additional fixed time called DIFS (DCF inter–frame space) and picks a random number between 0 and the current value of the back off counter CW (Contention Window) which is initially equal to 31. The picked value represents the additional amount of timeslots to wait if the medium stayed idle during the previous DIFS. The time slot is 20µ by default. So, the node keep waiting and sensing the medium until a transmission is launch by another node or its waiting time expires. If a transmission is detected (which should mean that another node has expired its waiting time and won the access to the channel) the node differ to the end of this communication and the number of timeslots to wait is decremented by the number of slots spent in waiting. This operation is repeated until the node wins. If so, the node starts the four–way exchange and if it fails its CW is doubled and the node restarts this process. CW will be reset to the minimal authorized value once an Ack is received (binary exponential backoff).

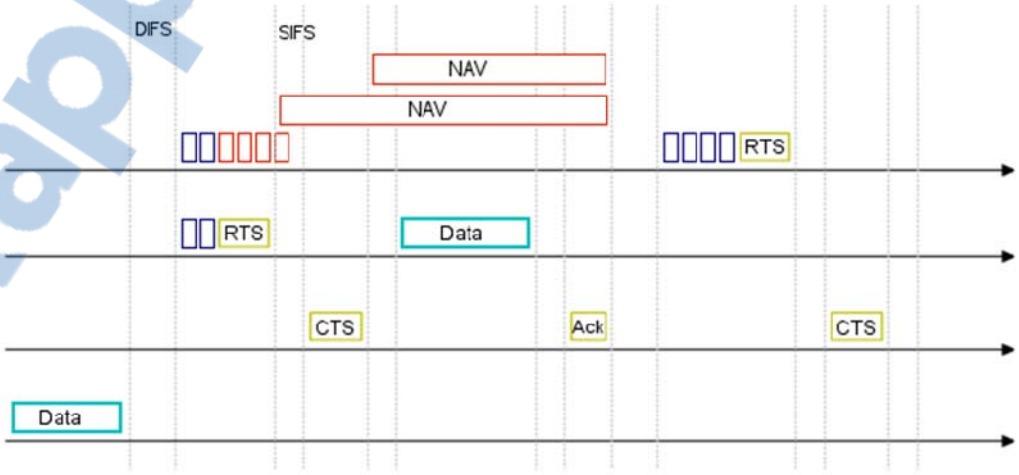


Figure 4-12: The DCF - four-way handshake.

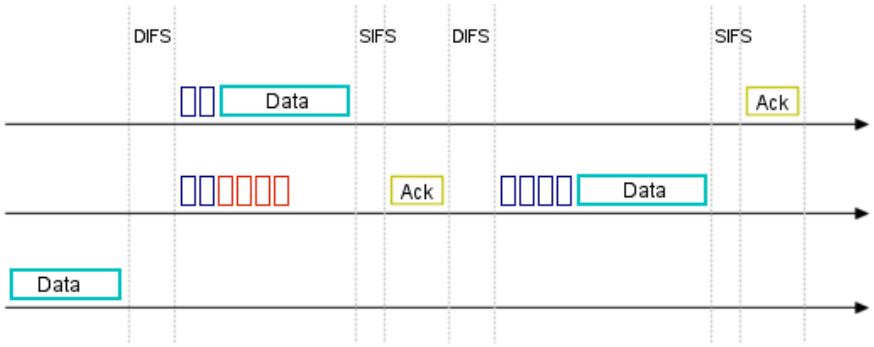


Figure 4-13: The DCF - the two-way handshake.

In the four–way mode a node needs to sense the channel idle for a DCF Inter–Frame Space (DIFS) interval before sending an RTS and a Short Inter–Frame Space (SIFS) interval before sending a CTS or an ACK packet. SIFS interval is shorter than the DIFS interval which gives priority to the Ack. If a transmission has been detected during the DIFS interval and it consists of an RTS (or a CTS), the node uses the information included in the packet to defer its transmission. This is referred to virtual carrier sensing. It is achieved by using time fields in the packets, which indicate the duration of the transmission in progress. This time field is called the Network Allocation Vector (NAV) field. All nodes that hear the RTS or CTS packets back off NAV amount of time before retrying again. In the case where the sensed signal is not understood, the node wait and sense the medium for a period equal to EIFS (Extended Inter Frame Space) which is longer than SIFS+CTS or SIFS+Ack unlike DIFS. Behaving this way allow the protection of possible CTS or Ack packets.

4.6. Conclusion

In this chapter we have tried to introduce the main issues and challenges in the design of MAC protocols for ad hoc networks and the common concepts behind the existing protocols in the literature. We have been able to see how it is difficult to design a MAC protocol which answers the requirements of today's applications. We think that designing an efficient MAC protocol depends on the offered possibilities by the physical layer. As well, it can require some interactions with the higher layers. This is because if looking at the layer level only few options are available and hence the possible solutions are delimited.

Quality is not an act, it is a habit.
Aristotle

Chapter 5

5. QoS Support in Ad Hoc Mobile Network

Abstract

Networking without QoS support can't be of big importance for today's applications and users. Therefore, QoS support in mobile ad hoc networks is an enforced choice despite the additional difficulty it comprises. This chapter presents the main works that aims to allow differentiated services in MANETs.

5.1. Introduction

Ad Hoc networks are very useful and promising. They allow achieving networking tasks without relying on any fixed infrastructure. They are to fixed networks what mobile devices to personal computers are. They can offer us the same feeling of freedom when using a tiny mobile device. However, they are still missing from our everyday life. Making ad hoc networks an active part of our being is facing many challenges and issues which are due to their inherent characteristics. The limited bandwidth, the varying link capacity and properties, the dynamic topology and the limited power source, all these factors make this class of networks very hard to apply. In the other side, today's applications are evolving and

call for more and more quality of service from the underlying networks. MANETs are also involved in this and must offer the most appropriate service to their users. This becomes an obligation with the actual trend toward the integration of the existing networks. Supporting QoS in ad hoc mobile networks is somehow paradoxical. At the same time as we complain of the lack of resources to achieve reasonable best effort networking, we try to add further loads. In fact, this is a right move because Ad Hoc networking can't be valuable without QoS support. However, it does not cancel that the mission has become very complex. In this chapter we will try to present the main achieved works in the direction of allowing QoS support for ad hoc mobile networks.

5.2. About Quality of Service

Yacine lives in Algiers, he is planning to visit Jijel and spend a few days of its vacation there. He can go by car and it will take him about 05 hours. But if he takes the plane this time becomes only twenty minutes. Both the plane and the car are supposed to take him to this city but offer two different services with different costs. This is the same with what is happening with traffic in computer networks. Some packets need to attain their destinations in less than a fixed time and some other packets are simply required to reach their destinations in a reasonable time. Therefore, researchers have begun to work on this problem. How to offer to each application the service it requires? In RFC2386 [62] QoS is defined as a set of requirements to be met by the network while transporting a packet stream from source to destination. Looking abstractedly to this picture, we can take out two important actors which are the application and the underlying system. The application has to specify its QoS requirements. This specification can consist of a set of values or intervals. The system should be able to determine if it is able to provide the specified service. If okay, it will eventually reserve the required resources to this service; otherwise it may simply reject the service request or negotiate with the application to fix new values. Several mechanisms are expected to monitor the accepted traffics and guarantee the agreed service such as scheduling and traffic shaping.

5.3. QoS in the Internet

The Internet is a packet switched network based on IP which was first designed to hold data traffic. The goal was to transport this data to its destination with all the offered resources without any constraints. That is what we call best–effort data delivery. However, with the evolution of the applications and their need and the all–IP perspective, such service becomes insufficient. The most common applications which are expected to meet a wide use on the internet are real–time traffics like voice on IP or Video on IP. These applications do not work well with BE (best effort) Internet and need some constraints to be respected. Therefore, researches have focused on how to deliver new services which answer applications needs like bounded delay or minimum bandwidth requirements. Network operators intend to offer a controlled end to end delay for real–time applications and also be able to control the bandwidth sharing between flows. Two QoS models have been proposed for the Internet, The Integrated Services model (IntServ) [61] and the Differentiated Services model (DiffServ) [63]. A QoS model does not define specific protocols. It only defines the overall architecture.

5.3.1. The Integrated Services model (IntServ)

The Integrated Services model has been proposed to extend the original Internet architecture in the direction of providing guarantees to applications. It integrates best–effort services, real–time services and controlled link sharing (hence the name). The key idea of the model is that guarantees can only be ensured via reservation. This requires that the network resources can be explicitly managed and hence some flows can be rejected if the guarantees they call for can't be answered (admission control). Such reservation is to be achieved for each flow, this forces intermediate routers to keep the state of all the flows that traverse them. It is also required that routers will be able to authenticate both users and following their flows' packets. Flows in IntServ might consist of one TCP connection or one audio/ video stream between a given host pair. It is the finest granularity of packet stream distinguishable by IntServ [61]. IntServ propose various QoS classes where two only have been specified, guaranteed service and controlled load service.

5.3.1.1. Guaranteed service

GS guarantees an amount of bandwidth along the route to the destination, a firm end–to–end delay and that no queuing loss will occur in the conforming traffic. This service is

dedicated for real–time applications which do not tolerate higher delays or data loss than an agreed threshold. This is achieved by reserving a wire of the bandwidth along the route and buffers in the intermediate routers to each admitted flow. Non conforming traffic will be treated as best effort traffic and eventually marked so that the next routers can recognize it.

5.3.1.2. Controlled load

This service is suggested for adaptive real–time applications which can tolerate higher delays but are sensible to network congestion. The aimed service is comparable to best effort service on a lightly loaded network. The difference is that with controlled load service the flows will not go down as the network load increase.

After the application makes clear its QoS requirements, the routers in the path to the aimed destination must be notified about these needs. Guaranteeing the service to the application is done via reservation of the needed resources. This reservation must be accomplished by all the routers between the source of the flow and its destination. A working group within the IETF has developed a resource reservation setup protocol called RSVP [64]. It is a signaling protocol which allows reservation setup and control. It is not expected to find the routes or replace the routing protocol; it only uses the routes supplied by the active routing protocol. The main message types in RSVP are the Path message, which is transmitted by the sender to initialize a new flow, and the Resv message, which is sent by the destination and achieve the resource reservations at the routers in the path. A Path message contains the field Tspec which defines the traffic characteristics of the data flow that the sender will generate. Tspec is to be used by traffic control to verify if the flow can be supported. It also includes the identifier of the session which consists of the sender identifier, the transport layer protocol and the port number. The Resv message is returned back on the same route. When receiving the Resv message, routers set the reservation if possible. Otherwise, an error message is returned back to the receiver. Once the reservation is accomplished, is must be refreshed periodically otherwise it expires and the reserved resources are released after a timeout. Additional messages are used to notify errors or to cancel the reservation.

IntServ/RSVP looks as if it is the straight solution to applications' QoS concerns. It appears that it can offer the required QoS for the most demanding applications but we can easily distinguish that this model cannot scale if the number of flows grows large (like in the

Internet) because of its per–flow care. Furthermore, the tasks to achieve for the operations of this model add big overhead on the routers when controlling and monitoring traffics. These facts make it reasonable to search for more approaches to treat the problem of QoS support in the Internet.

5.3.2. Differentiated service

This model suggests a new approach in QoS provisioning to overcome the shortcomings found with IntServ. This approach is based on flow aggregation and per hop behavior (PHB). In DiffServ flows are aggregated into a set of classes where each class is treated differently. The per–class view allows scalability of the solution in contrast to per–flow view adopted in IntServ. Core routers are discharged from traffic control tasks; they only perform forwarding according to the class of the traffic (PHB). The IETF has defined two types of per–hop behaviors, namely EF for expedited forwarding and AF for assured forwarding. EF is to be used for strict real–time traffics which require low delay and jitter, where AF is for traffics that do not have hard constraint in term of delay. Policing tasks are pushed to the edge of the domain. Thereby the overhead produced is less important comparing to IntServ. When an edge router receives a packet from the exterior it will classify it and fill the DSCP field by the corresponding value. This value is used by the core routers of the domain to decide about the PHB to perform. DiffServ do not offer a strict guarantee to each flow because it does not achieve per–flow reservation but it approximate it and the error relative amount can be insignificant.

We have tried to give a general idea of QoS provisioning approaches in the Internet. Of course, we have ignored many implementation mechanisms and details which are not relevant in our work. In the next session we will try to get a look on QoS in ad hoc networks.

5.4. QoS in MANETs

Ad hoc mobile networks are flexible networks and can be envisaged in many applications. However, this technology has many drawbacks which disturb its applicability. It stays very difficult to build a stable service on these networks because of their dynamic nature, mainly the dynamic topology, the varying link state and the limited capacity in term of

bandwidth and energy. In the other side applications impose new QoS constraints. The QoS constraints could be available bandwidth, end–to–end delay, delay variation (jitter) or the packet loss ratio. Despite these facts, researches give increasing attention to QoS provisioning in ad hoc networks. This is because it is what today applications call is for. The main works achieved in this direction attempt to give some adaptations at the different stack layers, to suggest new architectures or to evaluate and adapt the existing works used for conventional networks.

5.5. General models for QoS support in MANETs

A QoS model is a defined mechanism for achieving QoS as a whole. It does not define specific protocol but the general architecture, the components, the different functions and the relations therein. It may require interaction between the different stack layers as it may be designed by respect to the classical layered architecture. Next, we will overfly the proposed QoS models for MANETs in the literature. And compare between them and with the existing models for wired networks.

5.5.1. IntServ and DiffServ in ad hoc mobile networks

The unique characteristics of mobile ad hoc networks make it almost impossible to import existing solutions in other contexts. As sensed above, IntServ provides a per–flow service and allow applications to express their QoS requirements to the underlying network via a signaling protocol such as RSVP. The network resources are reserved to the admitted flows in order to guarantee the aimed QoS. The main drawback of such approach in the Internet is the incapacity to scale with the big number of flows. Dissimilarity, the number of simultaneous traffic flows in ad hoc networks is relatively low in most of the foreseen applications. Therefore, this problem can be of no weight in the ad hoc context but one can imagine that any reservation approach will be of a negative impact on the network performance because of the poorness of MANETs. The second drawback of IntServ in the Internet is the important overhead produced by the traffic policing and the signalization and undoubtedly this problem is more significant in MANETs. In the other hand, DiffServ resolve the problem of scalability by aggregating flows into a set of classes. It also discharges the core routers and pushes most of the traffic policing tasks to the entrance of the domain. In addition to that DiffServ basically do not require any signalization protocol. This allows

DiffServ to pass the drawbacks of IntServ and to be more appropriate to MANETs. However, DiffServ has been designed for Internet domains and it defines specific tasks to specific components which may not exist in MANETs. The concept of core router and boundary or edge router is ambiguous in the context of ad hoc networks since all the nodes are alike and should have the same responsibilities. A hybrid model has been proposed for ad hoc mobile networks in [65]. Called flexible QoS model for MANETs (FQMM), it combines some aspects from both IntServ and DiffServ to suggest a model for the ad hoc context with the advantages of these models.

5.5.2. Flexible QoS Model for MANETs (FQMM) [65]

According to the authors FQMM has a hybrid provisioning scheme that combines per–flow granularity in IntServ and per–class granularity id DiffServ and a relative and adaptive traffic profile to maintain consistent differentiation between traffic types and keep up with the dynamic of the network [65]. It is a DiffServ Adaptation for MANETs, by adding the per–flow aspect and by defining the boundary and core routers in the context of MANETs. DiffServ defines three kinds of nodes, the ingress node is the node which sends data and plays a similar role to edge routers in DiffServ domains, interior nodes are the intermediate nodes and egress node is the destination. This means that every node has a set of roles to play which depend of its position in the traffic path. Per–flow QoS provisioning is allowed for a limited set of traffics which have the highest priority. Ingress nodes are charged of traffic conditioning which sets the traffic profile. As mentioned above the traffic profile is relative and the treatment of every class / flow depends on the current network state. Reservations (implicit or explicit) are done in term of the percentage of available resources and not by fixed values. A preliminary simulation study has been realized by the authors shows that a differentiation can be achieved by this model but don't illustrate if it is possible to get suitable QoS guarantees. FQMM is unable to provide hard QoS guarantees because the adopted relativity in QoS provisioning. Indeed, application has some parameters to be within some intervals to operate properly; these parameters depend only on the application and are independent from the available resource on the network at any time therefore if the network becomes unable to satisfy them the application will crash.

5.5.3. Stateless Wireless Ad hoc Networks (SWAN) [66]

SWAN is a model which is stateless and dedicated for services differentiation in ad hoc mobile networks. It was developed by the Comet team at Colombia University. The key idea behind the model is to provide suitable service for real–time traffics through local control of best effort flows. Admission control is done in the sender on the basis of bandwidth estimation along the path to the destination. Admitted flows regulation is done on congestion detection by explicit congestion notification declared by the intermediate nodes. The main characteristic of SWAN is its flexibility and simplicity. It does not require any complex tasks such as signaling or state control mechanisms to keep on real–time sessions. This of course can not allow a strict control of the flows and resources in the network however it saves big efforts and reduces the resulted overhead in a highly dynamic environment such as MANETs. SWAN includes a number of mechanisms as illustrated in fig. 5.1. The classifier splits the incoming traffic into real–time traffic and best–effort traffic. Best–effort traffic is processed by the shaper which consists of leaky bucket which is controlled by the rate controller. The react controller reacts to the delays recorded in the MAC layer. Admission control is done locally after achieving a probe. The next sections describe the main control algorithms of SWAN which are rate control, admission control and traffic regulation.

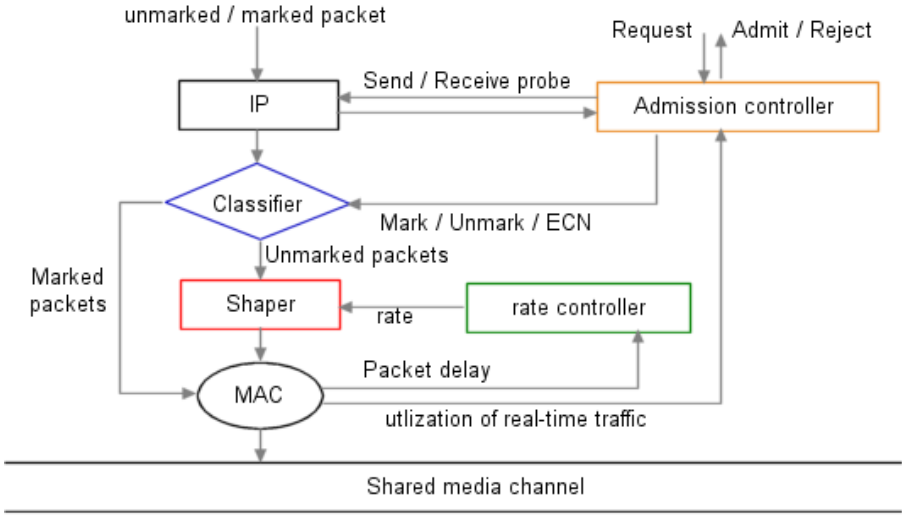


Figure 5-1: the SWAN model architecture [66].

5.5.3.1. Rate control of best effort flows

Controlling the rate of best–effort traffic means that best–effort flows can only occupy the resources which are not imperative to real–time traffic. The local rate controller defines the rate at which the best–effort traffic is transmitted in function of the delays induced by packets at the MAC layer. SWAN does not necessitate QoS aware MAC protocols. Packets delays are measured within each period T and if one or more delays pass a fixed threshold the controller respond by reducing the rate otherwise it will get increased. The adopted algorithm is called AIMD [67] for additive increase and multiplicative decrease.

The authors justify the use of packet delays feedback to adjust the best–effort transmission rate by a comparison to an analysis done in [67] that explains the delay and the throughput in function of the network load in a congestion controlled system (fig. 5.2). The graph shows that the load at the delay "knee" is the best option since the delay is very low and the throughput is about equal to the highest possible value. The SWAN AIMD algorithm is expected to keep the system at this load threshold.

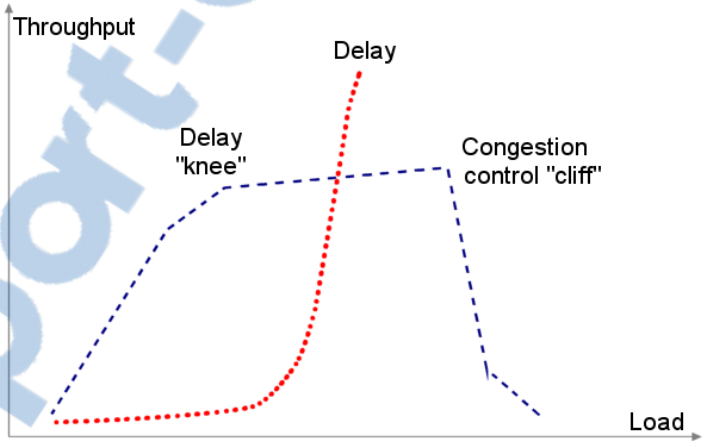


Figure 5-2: general behavior of a congestion controlled system [67].

5.5.3.2. Admission control

The admission decision is taken in the sender after probing the nodes on the path to the destination. A node which aims to send real–time traffic sends a probing request packet toward the destination to learn the bandwidth bottleneck on the path. The sent packet contains a bottleneck field to be set by the nodes on the path. When this packet reaches the destination, the probe response is returned to the sender and allow the admission controller

there to decide about to accept the traffic or reject it. If a real–time traffic has been admitted the packets will be marked as RT by the classifier. The available bandwidth at each node is estimated by listening to the real–time traffics sent on the channel.

5.5.3.3. Real time traffic regulation

The admission control technique adopted in SWAN is very simple and it is not well tuned therefore it may happen that some traffic gets admitted while no resources are available to keep it. This can occur if two or more traffic send the probing request simultaneously toward a node. The node will set the same estimation of the available bandwidth on all the probing packets which make it possible that the traffics will get admitted to use the same resources simultaneously and cause congestion at this node. Another case is when route between to nodes change due of the dynamic of the network. In this case if there were a few real–time traffics admitted on the first route they will be automatically rerouted on the new route without knowing if there are enough resources to hold them and this may cause congestion some node of the new route. SWAN adopts a dynamic regulation of real–time traffics to solve this problem. When congestion is experienced in a node a congestion indicator bit is set in the real–time packets that pass. Any destination that receive such ECN (explicit congestion notification) will send a regulate message to the sender. The sender at its turn must reestablish the session by a new probing request. Of course this operation must be done carefully to avoid that many probing requests will be sent simultaneously. Random delays can be adopted here to avoid stopping many sessions at once.

The key idea in SWAN is to achieve lightweight tasks in order to support real–time services rather than adopting a complex architecture that may control such a dynamic environment. SWAN do not try to get a full control on the network, it simply admit that ad hoc networks are dynamic environments which are unpredictable and tries to deal with this fact to allow the intended services with low costs. Therefore, SWAN will not offer strict QoS guarantees.

5.5.4. INSIGNIA [68]

The INSIGNIA QoS framework has been designed to support adaptive services in MANETs. It is based on an in–band signaling technique plus a soft–state approach in

resource management. Soft–state resource management here stands for reserving resources for short periods rather than keeping the needed resources for all the session like in virtual circuits. This approach fits very well to MANETs in contrast to hard–state approaches. When adopting soft reservation states these reservation must be refreshed constantly through the user session. The second key element of INSIGNIA is in–band signaling which refers to the fact that the control information is carried in data packets. Using such approach allows faster response to topology changes and less control overhead. The INSIGNIA framework does not achieve routing. Instead, it relies on any presented protocol to determine the needed routes. However, its performance highly depends on the speed at which this protocol can find new routes or restore the lost ones. The general architecture of the INSIGNIA QoS framework is shown in fig. 5.3.

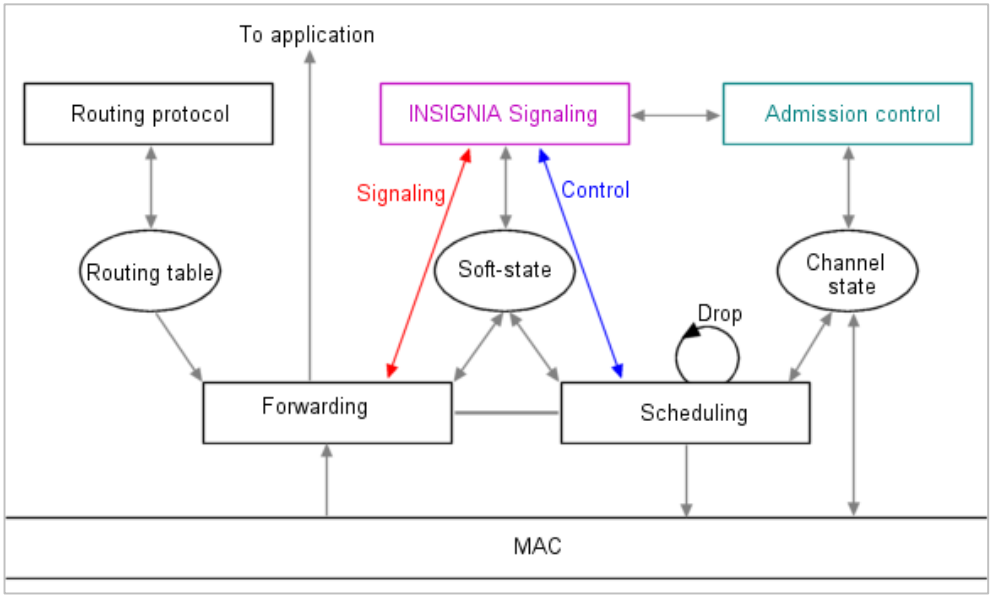


Figure 5-3: Architecture of the INSIGNIA framework.

The main component in the INSIGNIA framework is its signaling system. Many documents refer to INSIGNIA as a signaling protocol rather than a QoS framework. The INSIGNIA signaling system purpose is to establish, adapt, restore and terminate end–to–end reservations. As mentioned above the framework adopts in–band signaling. An IP option field is used for that as illustrated in fig. 5.4.

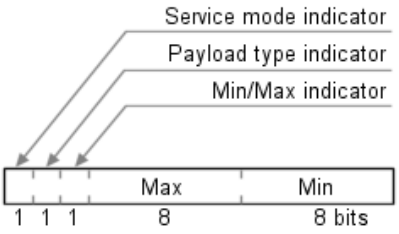


Figure 5-4: INSIGNIA IP option.

The service mode indicator is a 1 bit field which takes two possible values RES or BE. RES is set by the sender to specify that the application therein require a differentiated service. The payload type indicates the type of the packets. It takes two different values EQ and BQ. EQ refers to enhanced QoS and BQ for base QoS. With the INSIGNIA framework applications can choose two level of operation to allow adaptive services. The application benefit form the enhanced QoS if the necessary resources are available otherwise and if only the base QoS can be guaranteed the application adapts to the base QoS level. These two levels are defined by the max and the min bandwidth in the INSIGNIA IP option. However, the meaning of the signaling message depends on all the values in the INSIGNIA IP option. The min/max indicator shows what is the service level claimed or allowed. In the sender it is set to the required service level, max for EQ and min for BQ. In the intermediate nodes or at the destination this field shows what the reserved level in the upstream nodes is if the service mode is still set to RES. The main scenarios in the protocol are:

5.5.4.1. The reservation / restoration

This operation starts at the sender. The packets IP headers of the flow which are intended to receive a differentiated service are expanded to hold the INSIGNIA Option. The service mode is set to RES and the min/max indicator is set to the needed reservation. If set to max the intermediate nodes will achieve the reservation of the maximum value mentioned in the field Max. Otherwise, the minimum is used. The payload indicates the packets type. At any intermediate node, if receiving a RES/MAX and only the minimum bandwidth can be reserved, the indicator is set to MIN and the minimum bandwidth requirements are reserved. In addition to that if the packet is tagged as EQ it will be degraded to best–effort level by changing the service mode to BE. This means that downstream nodes will not continue the reservation process and only a partial reservation has been done between the source and this

bottleneck. The application can choose to release or keep these resources after being informed by the reports of the adaptation mechanism resident in the destination. The choice depends on the nature of the application. After the packets reach the destination, it sends a report to the source to inform it about the achieved reservation. The same mechanism for reservation will be useful for restoring resources after a route change.

5.5.4.2. QoS reporting and the adaptation

When the reservation packets reach the destination, the destination will learn the state of the route. Receiving RES/BQ/MAX means that the application at the sender is now operating in BQ level and tries to upgrade the service by attempting to reserve the maximum. The MAX indicator has reached the destination which means that the reservation has been done along the path. The destination will report this to the sender which will probably switch to EQ service. The reception of RES/BQ/MIN does not add important information since the application is operating in the only available level. May be the sender has sent this same combination or it has sent RES/BQ/MAX and some bottleneck on the path switched the indicator. In this case some partial max reservation may exist and reporting this information to the sender can be useful to release the extra reserved resources. In the other hand, receiving BE/EQ/MIN means that the enhanced QoS level can't be guaranteed and reporting this information to the sender can push the application to adapt to the base QoS level. While receiving BE/BQ/MIN means the only best effort traffic is allowed for this flow. QoS reporting is achieved periodically or when necessary in response to an event that occurs. It depends mainly on the nature of the application and its sensitivity to some events.

The key contribution in the INSIGNIA QoS framework is its in–band signaling system. The INSIGNIA signaling system permits to reduce the bandwidth consumed by the control packets by adopting in–band signaling. The performance of the framework depends highly on the other protocols that serve it. INSIGNIA do not specify any routing or MAC protocols but with its approach a QoS–aware MAC and adequate routing protocol are necessary to its performance. INSIGNIA is not a generic QoS framework; it tries to support only one class of applications which are adaptive applications like VoIP and video on IP. It lacks some flexibility in the QoS requirements specification by only offering two values, the maximum and the minimum. This can be okay to some applications but one may think that if the upper level cannot be satisfied, it

is not obligatory to switch to the lowest level. Instead an appropriate level can be taken between the minimum and the maximum level.

5.6. QoS support from a layered perspective

Following, we examine the QoS provisioning issues and the achieved work from a layered perspective. We have voluntarily ignored the physical layer because it not relevant to our subsequent work.

5.6.1. QoS provisioning at the MAC layer

The MAC layer is very important in mobile ad hoc networks and has imperative role comparing to other networks classes. It is practically impossible to provide end–to–end QoS without tuning the MAC layer for this purpose. Most of the QoS supporting components at the upper layers assume QoS aware MAC protocol. Therefore, it must provide efficient use of the available resources while satisfying the applications QoS requirements. MAC layer should also provide good estimation to the available resources and achieve adequate scheduling to optimize delay. The need to centralized control becomes important when referring to QoS provisioning and resources reservation. However this is not feasible in ad hoc networks so alternative techniques must be explored to this end. We can classify the main achieved works for QoS provisioning at the MAC layer into two main approaches. In the first approach the network is organized into clusters and some nodes play the role of the access point in centralized wireless networks. TDMA is used within a cluster, a technique that requires synchronization between the comprising nodes. The second approach use completely distributed techniques and do not require any synchronization which is more fitting to this networks' class. Next, we will present important achieved work in this direction.

5.6.1.1. Real Time MAC [69]

It is a variation of the IEEE 802.11 protocol which supports real time traffic by enhancing collision avoiding and avoiding the transmission of already expired packets. To achieve this, each packet is associated with a deadline RT–MAC scheme uses a packet transmission deadline. In the enhanced collision avoidance scheme the next backoff is sent in the current transmission which allow neighbors to learn about it and thus avoid collision with the next transmission of that node. Real time packet will get dropped if they expire

before the gain the access to the medium. This can save important bandwidth resources. The sending node checks whether a packet has expired at three points: before sending the packet, when its backoff timer expires and when a transmission goes unacknowledged. RT–MAC achieves a more efficient service that the original IEEE 802.11 but this does not make it a QoS aware protocol.

5.6.1.2. IEEE 802.11 DCF with priority classes [70]

This is another variation of the IEEE 802.11 protocol that allows priority access for different classes of traffics. It supports many priority classes and the prioritization is achieved by using a combination of shorter IFS or waiting times and shorter backoff time values for higher priority traffics. The length of the IFS and the backoff windows determines the priority of the traffic in hand. Using simulations, the authors have shown that this variance has better performance than the original one in terms of throughput, access delay and frame loss probability for real time traffic. However, this scheme mishandles low priority traffic which accumulates higher delay due to a longer backoff time even if no higher priority node is transmitting which causes bandwidth to be wasted.

5.6.1.3. Enhanced 802.11 DCA (EDCA) [72]

IEEE 802.11e [72] was proposed as an extension to the IEEE 802.11 MAC in order to provide service differentiation the answer the potential need of supporting real–time audio/video in WLANs and MANETs. The 802.11e introduces the Hybrid Coordination Function (HCF), which defines two new MAC mechanisms. The HCF controlled channel access (HCCA), and the enhanced distributed channel access (EDCA).

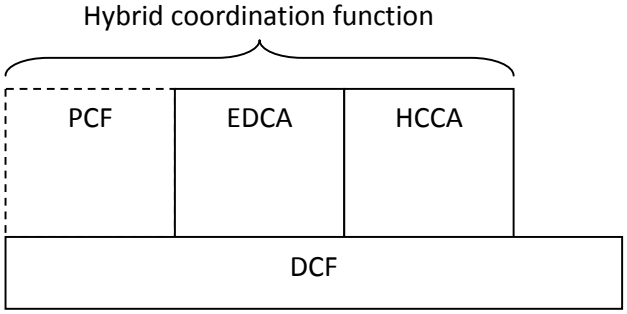


Figure 5-5: 802.11e MAC layer architecture [72].

The DCF remains the fundamental access method used by non–QoS traffic and the PCF is optional (see fig. 5.5). The PCF is to be used for contention free non–QoS nodes in the (centralized mode) otherwise it is optional. The EDCA is what we are interested in because it does not require central administration just like 802.11 DCF mode and allows prioritized QoS services (see fig. 5.6). HCCA also supports QoS but with a completely different approach (called in the standard specification parameterized QoS) which, like the PCF, requires a centralized architecture.

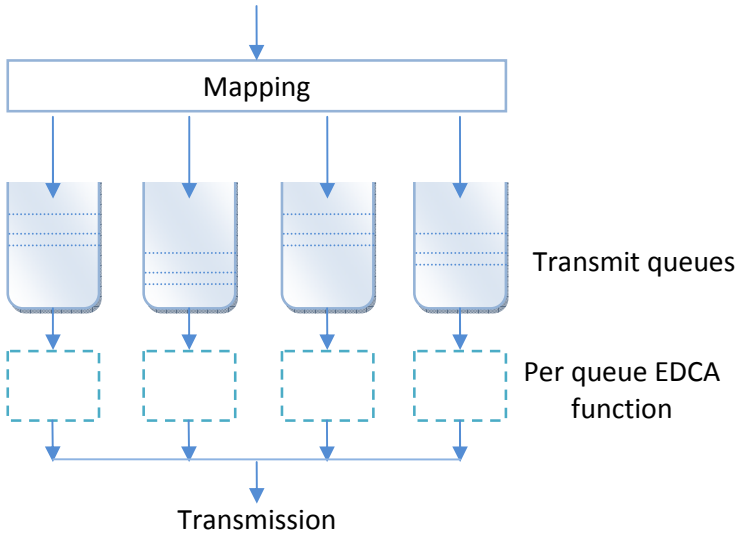


Figure 5-6: EDCA access categories.

EDCA achieves service differentiation by introducing four different access categories (ACs), namely voice, video, best effort and background traffics [72]. EDCA uses a separate transmit queue for every AC, which are contending for accessing the medium. Every AC has different parameters (backoff, CW) which are chosen in a way that relative priority is given to more important ACs. The AC with the smallest backoff wins the internal contention. Collisions between contending frames within a same node (called virtual collisions) can occur when the backoff counters of two or more AC reach zero at the same time. These collisions are solved by giving access to the AC with the higher priority. The other ACs behave as in case of external collisions. In EDCA, the equivalent of DIFS we have previously seen in 802.11 DCF is called Arbitration Interframe Space (AIFS). Both AIFS and the contention window size are dependent on the AC (AIFS[AC], CW[AC]). The service differentiation is sufficiently achieved because high priority traffic will use relatively a smaller AIFS[AC] value as well as a

smaller CW min[AC]. Thereby, higher priority queue always defers less time before attempting to transmit. So, when multiple priority queues content for channel access, higher priority queue is likely to seize the channel more that less priority traffics.

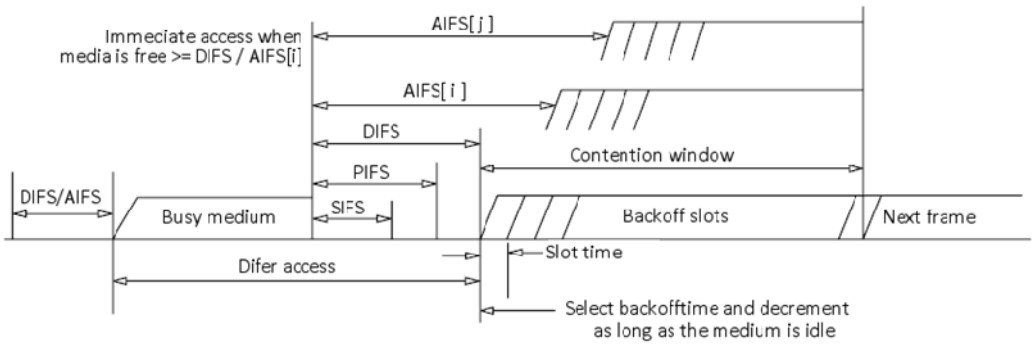


Figure 5-7: Interfame spacing / IFS in IEEE 802.11e [72].

It is important to note here that many changes have been introduced to the IEEE 802.11e standard since its introduction. The description given here is based on [72].

5.6.1.4. Black Burst contention [71]

BB contention is an extension to CSMA/CA MAC protocols, like the IEEE 802.11 MAC, which try to provide delay–bound guarantees in mobile ad hoc networks. Nodes contend for medium after the medium has been idle for a period longer than inter–frame space where nodes with best–effort traffic and nodes with real–time traffic use different inter–frame space values. This gives priority to real–time traffic. When the medium remains idle long enough (After DIFS in the IEEE 802.11 DCF), real–time nodes contend for transmission by jamming the medium with pulses of energy (black bursts) where each contending node is using a BB with different length. The length of each BB is an increasing function of the contention delay experienced by the node, measured from the instant when an attempt to access the channel has been scheduled until the node starts the transmission of its BB. At the end of the BB, a node senses the channel for a fixed duration to make out if there is any ongoing BB or not. If it finds the medium idle it wins the access to the medium. This winner is the one that experienced the longest delay. Nodes that lost the contention will join next round of contention with new longer BB's. BB contention enhances collision avoidance and solves the packet starvation problem. It ensures that real–time packets are transmitted without collisions and with priority over best–effort packets. The BB contention scheme thus

provides some QoS guarantees to real–time traffic in comparison with simple carrier sense MAC protocols. However, BBC, like other CSMA/CA protocols, does not fully solve the hidden nodes problem.

5.6.1.5. MACA/PR (Multihop Access Collision Avoidance with Piggyback Reservation) [73]

MACA/PR defines an architecture which provides guaranteed bandwidth support by reservation to real–time traffic. It is based on MACAW [50] and composed from three main components, a MAC Protocol, a Reservation protocol and a QoS routing protocol. We will not care about the routing algorithm here. The reservation along the entire path is made by the first data packet (piggybacking) in the real–time stream. A RTS/CTS dialog is used on each link for only the first packet where both RTS and CTS specify how long the data packet will be. Nodes which hear the CTS will avoid colliding with the following data packet. The RTS/CTS dialog is used only in first packet to setup reservations. The subsequent packets do not require this. When sender sends a data packet, the sender schedules next transmission time after the current data transmission and piggybacks the reservation in the current data packet. The receiver keeps the reservation in a table RT (reservations table) and confirm with an ACK. The neighbor nodes which hear the data packet can learn about the next packet transmission time and at the receiver side, the neighbors hear the ACK and should avoid sending at the time when the receiver is scheduled to receive next packet. If the ACK can't be received many times, he link is assumed to be not satisfying the bandwidth requirement.

We'll not detail further in this section, we described in small detail some works that may help us in the coming part of the document.

5.6.2. QoS routing

Many routing protocols have been proposed for mobile ad hoc networks. Often classified into reactive and proactive schemes, many techniques have been conceived to deal with the unique characteristics of these networks. However, there is no absolute convergence on a set of protocols that will be used in real application. The MANET group of the IETF is working on the standardization of routing protocols for MANETs. Some protocols have been standardized like Ad Hoc On Demand Distance Vector (AODV) [23], Optimized Link State Routing Protocol [22], Topology Dissemination Based on Reverse–Path Forwarding (TBRPF)

[25] and The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4 [19]. Some other protocols specification drafts have expired like TORA [17], CEDAR [74] and FSR [75] and others are currently active like OLSRv2 [78] and DYMO [34] which has been recently developed by the IETF MANET group. However, these protocols do not consider the QoS requirements of the applications. QoS routing does not consist of only finding optimal routes. It must find routes that satisfy the application requirements like minimum bandwidth, maximum delay or maximum loss ratio. This task is more difficult in ad hoc networks therefore the network layer for MANETs should receive special care. Designing QoS aware routing protocols has taken two directions. First, new QoS routing protocols has been built by adding QoS measurements to the existing QoS–Ready protocols like AODV (QoS–AODV [76]) and DSR. The second direction consists of rethinking new protocols that offer QoS support like CEDAR (74), TBP [77]. These protocols are QoS aware and the routes determination is done with the QoS in head.

5.6.2.1. CEDAR (A Core–Extraction Distributed Ad Hoc Routing Algorithm) [74]

CEDAR has been proposed for small and medium size ad hoc networks. “The protocol tries to provide route that are highly likely to satisfy the bandwidth requirements [74].” The protocol dynamically establishes the core of the network and exchange the topology updates within it. CEDAR has three key components: a) the establishment and maintenance of a self organizing routing infrastructure, called the "core", for performing route computations, b) the propagation of the link–state of stable high–bandwidth links in the core, and c) a QoS route computation algorithm that is executed at the core nodes using only locally available state.

The basic concepts behind the design of CEDAR are inspired from the problems found in the existing solutions. An important problem found in link state protocols is the big overhead produced in the establishment and maintaining of the global state at each node. Link state protocols allow all the nodes to have the global state of the network and to refresh this state regularly. Having such information, each node can compute the route to any destination but only few routes are needed in the entire network. CEDAR proposes to reduce this load by limiting the propagation zone of the links states. The size of these zones is a function of the capacity of the related link. Thereby, a bad link can only be seen by close

nodes in contrast to a good link whose the state can reach farther nodes. In this way every node will have a different state which is centered in this same node and becomes less dense as moving away from it. CEDAR also proposes to limit the number of node involved in the state propagation. Another important problem in the perspective of CEDAR is the adoption of unreliable broadcast to flood packets in the network. CEDAR proposes an alternative mechanism which is based on Unicast transmissions.

a. The core construction

The core of the networks consists of a dominating set which must be as close as possible to the minimum dominating set. However, finding the minimum dominating set is an NP–hard problem. CEDAR proposes a simple and effective distributed algorithm to construct a dominating set which is approximates the MDS. In this algorithm the nodes periodically send a beacon which contains some information on the node like its degree and its effective degree defined by the number of neighbors which chose it as a representative in the core (their dominator). This information helps the node to choose their dominators. When a node choose a dominator it immediately sends him the list of its neighbors and their dominators and if a node is elected to join the core (chosen as a dominator by some node) its piggyback in its beacon a message which contains its ID and a field to fill the path traversed from this elected node which is initially set to null. This message is to be piggybacked by all the neighbors and the path is updated until it reaches all the nodes after three hops. The goal is to inform the nearby core members and set virtual links with them (see fig. 5.8).

b. The links states propagation

CEDAR uses increase and decrease waves to propagate the state information in the core. We have mentioned above that in CEDAR only a subset of the network comprising nodes (only the core nodes) take part of the state propagation operation. The propagation of the waves is done by a specific flooding mechanism in the core. This mechanism is based on Unicast transmissions to achieve reliable operation and eavesdropping and caching RTS/CTS packets optimal flooding. Increase waves propagate slowly and carry the link amelioration information. At the other hand, decrease waves propagate faster and carry link departure or deterioration information. Of course a threshold is needed to decide about to propagate the wave to avoid state fluctuations in the network. Each wave has a maximum distance it is allowed to traverse. Low bandwidth increase waves are allowed to travel a short distance,

while high bandwidth increase waves are allowed to travel far into the network. This distance is an increasing function of the available bandwidth. Thereby, stable high bandwidth link state will propagate throughout the core, and low bandwidth and unstable link state stays local.

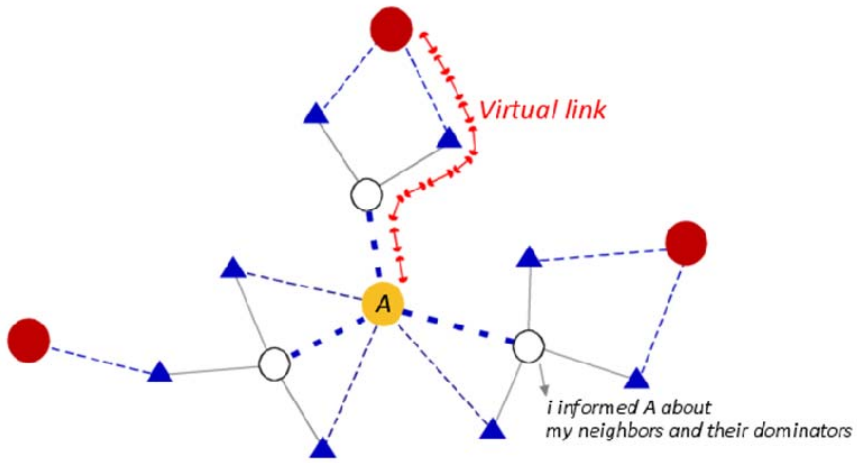


Figure 5-8: The topology information learned by A as a member of the core by beaconing.

c. The routes computation

QoS route computation in CEDAR is done on–demand. When a source node seeks to establish a connection to a destination node it asks its dominator by sending the tuple (source, destination, and bandwidth). The core node has dense local state information just by means of beaconing (see fig. 5.8). In addition to that, the link state propagation adds more information about further nodes in the network. This state is incomplete but may be sufficient to compute admissible routes to many destinations but not all the possible destinations. If the core node is capable to compute the route from its local state it will send it immediately to the demander. Otherwise, if it has a core path to the dominator of the destination it starts the route establishment phase. If not, it must first set up the path to the dominator of the destination. This is done by a request / reply discovery technique using the flooding mechanism employed in the waves propagation. Once the core path is established it is used by the dominator of the source as a guide to compute the needed route. The dominator of the source uses the information it owns to find an admissible path from the source to the

domain of the furthest possible core node according to the core path. This core node will do the same thing and so on until reaching the destination.

CEDAR tries to perform routing with lowest costs comparing to link state protocols by limiting the propagation of the links states changes. The range of this propagation depends on the significance of the change and the available bandwidth on the link. In addition to that only a subset of the nodes takes in charge the propagation of the state information in the network and the routes computation. This lightens the load on regular nodes as much as it makes it heavy on the core nodes. It is important to remark here that nodes that have big degrees are more candidates to join the core. Having a big degree means being exposed to high contention and joining the core means needing more resources to achieve the added responsibilities. This is somehow contradictory and can affect the performance of the protocol. In addition to that in CEDAR every link state needs to be propagated alone in contrast to link state protocols where all the links of the node are sent in the same update. This produces more packets and thus more communication over head. Therefore, the performance of the protocol can only be judged after thorough simulation or real testbeds. Applications can only specify their bandwidth requirements. We think this is sufficient since bandwidth and delay are correlated parameters. Using multiple metrics induce more complexity and affect the performance of the protocol.

5.6.2.2. Ticket based probing algorithm [77]

TBP approach is proposed as a general QoS routing scheme, which can handle different QoS constraints [77]. Nodes keep local state about the outgoing links which includes the delay of the link, the available bandwidth and its cost. TBP differentiates between stationary links and transient links. The links between the fixed or slowly moving nodes are likely to exist for long time. Such links are called stationary links and the links between the fast moving nodes are likely to disappear quickly. Such links are called transient links. The cost metric can be used to favor stationary ones. The end–to–end state is established by mean of distance vector algorithm and this state is inherently imprecise in a dynamic environment such as MANETs. Therefore two additional metrics are added which consist of the delay and the bandwidth variations. The algorithm achieves a multi–path

probing method; One probe for each path. In each probe p, the states of probing are recorded, including the traversed path, the accumulated delay and the accumulated cost. Thereby, the protocol offers multiple admissible paths to the any destination. Ticket based probing is similar to flooding algorithm but it is limited by the number of tickets generated at the source and guided by the states in the intermediate nodes. It is equivalent to flooding if the number of tickets is infinite. The information at the intermediate nodes, both local and end–to–end states, is collectively used to direct the probes.

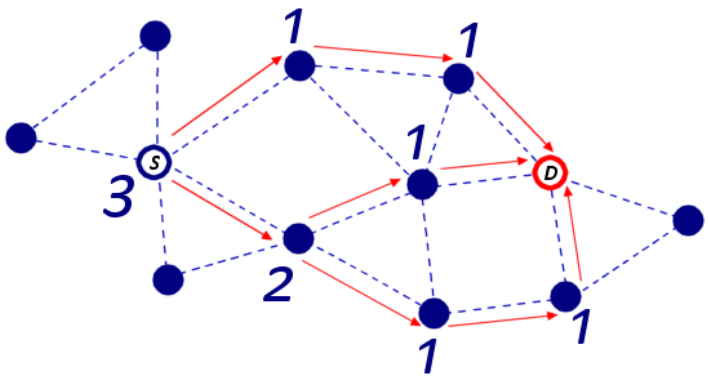


Figure 5-9: Discovery split and Tickets distribution.

A probe may contain multiple tickets; at an intermediate node, a probe with more than one ticket is allowed to be split into multiple ones – as long as there are sufficient tickets – each searching a different downstream sub–path. The maximum number of probes at any time is bounded by the total number of tickets. Since each probe searches a path, the maximum number of paths searched is also bounded by the number of tickets [77] (fig. 5.9). There are two kinds of tickets: green tickets and yellow tickets. Probes with green tickets are supposed to follow paths with less end–to–end cost. Where probes with yellow tickets seek optimizing end–to–end delay. At the destination node, the incoming probes will be collected. The routing overhead is controlled by the number of tickets generated which at its turn depends on the required QoS and the level of imprecision in the end–to–end states kept in the intermediate nodes. A big ticket number increases the chance of finding a feasible path and thus helps to tolerate information imprecision.

5.6.3. Transport layer in MANETs and QoS

UDP and TCP are the two transport layer protocols widely used by the applications. UDP do not react to congestion and can easily overwhelm the network with data, which wastes big network resources especially in a poor environment such as MANETs. Therefore, some techniques should be investigated to lighten this issue. In contrast, TCP has an inherent congestion control scheme and without a doubt has a great contribution in the success of the Internet. However, TCP has been first designed for wired networks where the packets losses are in the most cases due to congestion. TCP has been design on the assumption that losses are only caused by congestion. In ad hoc mobile networks the loss ratio is important and in most cases the links degradation or departure and the interferences are the cause of this loss. TCP misinterpret wireless errors as congestion. Applying congestion control and avoidance adopted in TCP for ad hoc networks result in very low end–to–end throughput.

Many techniques have been proposed to cop with this problem. Some techniques try to hide non–congestion losses from the sender by using local ACKs [79] or to only make the sender aware of the existence of wireless hops in the path so that it can behaves carefully. Another proposed strategy consists of adding Explicit Loss Notification (ELN) option to TCP acknowledgments or using ECNs (Explicit Congestion Notifications). In all cases, the performance of the transport layer depends highly on the adopted techniques in the other layers of the stack and mainly the lower layers. For example the MAC layer ACKs permit to reduce the end–to–end losses and help TCP to perform better. In addition to that information available at the lower layers is useful in the transport layer to decide about the action to take in front of some conditions.

5.6.4. Application layer

The application layer has an important role in supporting QoS in MANETs. The application in ad hoc networks should be more flexible. MANETs are dynamic networks and require reactive and adaptive applications especially for real time multimedia applications. Application has to adapt their parameters to the varying conditions of the network. For example, it is possible to envisage that application change the coding parameters or the compression level if the available resources change significantly.

5.7. Conclusion

Supporting QoS in mobile ad hoc networks is a very difficult. Many considerations must be taken at all the layers. But also interaction between the layers become suitable and may be indispensable. The classic wire–line centric network design must be revised. It is important to sense here that since ad hoc networks are unique environments, the envisaged application should be different than what we had in classic networks. Claiming the same applications used in the internet for ad hoc environments is a lot irrational. We think that hard–QoS constraints can't be answered in MANETs. In the same time we are convinced that ad hoc networks stay practical and can offer QoS to applications but this must involve all the components of the network and may be additional interaction even with the user who should evaluate QoS differently and what he expects in MANETs should not be the same what wired networks are capable of. The next chapter presents a new design approach which aim to exploit additional possibilities by adding extra interactions. We mean the cross layer design approach.

You can never solve a problem on the level on which it was created.
Albert Einstein

Chapter 6

6. CROSS LAYER DESIGN OF AD HOC MOBILE NETWORKS

Abstract

The performance of applications in ad hoc networks is affected by their inherent limitations. The answer the needs of those applications, researchers have turned toward the architecture. Lately many proposals suggest new and non regular interactions between the different layers. This chapter illustrates the concept of cross layer design and investigates how far can this approach remains beneficial.

6.1. Introduction

Wireless communications have known a big proliferation today and wired network is no more the default option in a big part of today's networking applications. All IP networks today are based on the same architecture which is the TCP/IP architecture. This architecture is based on the layered open systems architecture for networking. However, this architecture has been designed for wired networks and internets. Wireless links are far different from

wired ones in term of reliability and stability. They also have unique characteristics, the expected signal quality of a wireless communication link is relatively lower, less stable, and less predictable comparing to wired link. This pushes us to ask an important question. Does the TCP/IP architecture allow an efficient utilization of wireless networks? Also, does this architecture allow reaching or approaching optimality by the built–on algorithms? In reality nothing can help to affirm that the answer will be yes. Furthermore, running the layered TCP/IP architecture in wireless networks shows that it lacks of the needed flexibility to cop with the varying conditions of wireless links especially in applications where a QoS is required. This is why in the last few years many proposals suggest to violate the classic layered architecture as a way to reach optimal performances in wireless networks. These proposals rely principally on adding new interfaces or on design coupling of the stack layers. Such design termed in the literature cross–layer design is the subject of part of the document. We are interested in cross layer design to optimize applications performances in ad hoc mobile networks. This chapter aims to introduce the concept, to determine if we really need to follow this direction to answer the needs of today's applications and to investigate the new problems that can arise.

6.2. What is CLD?

Layering is the technique actually used to achieve networking. The networking functions are partitioned into a hierarchical set of layers where each layer achieves a subset of these functions. Each layer benefits of the services provided by the layer just below it, and it provides service to the layer above it. The communication is limited to only between the adjacent layers and with a determined set of primitives (fig. 6.1). This technique allows dividing the global task into more simpler and independent modules where each one has determined terms to fulfill. When working on a layer the designer should respect these terms and respect the interactions specified in the layered architecture. Otherwise, he is about to perform a cross–layer design. So cross layer design is a novel design principle where the key idea is to exploit the offered functionalities by the original layered architecture but also allow new interactions and joint design of protocols crossing different layers. Cross–layer design is suitable for specific scenarios, such as wireless networks, where the layered restrictions limit the reachable performances. It violates the specification of the layered architecture and this violation can take many forms like adding new interactions between layers by adding new

interfaces or sharing information between layers, merging layers or jointly designing two or multiple layers which are independent at the origin. Cross–layer signaling is also a key element of cross–layer designs when seeking the compatibility with the old architecture. The goal is to drive the required information to its intended destination without creating new interfaces. The packets headers can be exploited for this purpose as is the case with ECN [85]. The ICMP [90] message also has been used to propagate information across layers for a similar end. An ICMP message can be generated if a network parameter changes to notify any remote stack layer.

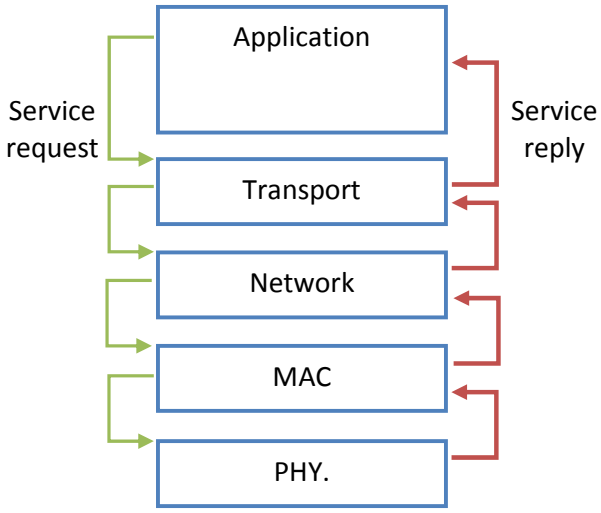


Figure 6-1: The layered architecture.

6.3. Motivation for CLD

The concept of cross–layer design has appeared with the manifestation of wireless communications. Initially, the design of wireless networks has been under the impact of the success of the layered architecture in wired networks. Therefore, designers only adapt the existing designs to wireless networks without thinking about any architectural modification. However, the layered architecture as used for wired networks is not necessarily the adequate architecture for wireless networks since it has been designed for a very different context. This pushes researchers to think if it is necessary to violate or completely rethink the classic architecture. Indeed, many factors go into making of this choice the best available option. First, the differences between wired and wireless networks are important and do not tolerate

abstraction. The link in wired networks is determined by a wire between two nodes, it is almost stable and reliable. At the opposite, for wireless networks a link is not stable and it is determined by a set of factors like distance, transmission power and the environment. The signal to interference and noise (SINR) at the receiver determines if the link is viable or not. In addition to that wireless communications are broadcast by nature which adds new concerns and opportunities. Experiments also confirm that the layered architecture is not suitable for wireless and experiences many limitations. The problem becomes more complex for ad hoc mobile networks where no centralized administration exists and the dynamic of the topology is high and unpredictable. The designer of ad hoc networks is expected to offer high performances to the increasing demands of today's applications while he has a small set of means in a poor environment. This designer has only one solution which is to efficiently exploit the available resources the maximum possible and will absolutely find himself forced to create new interactions to get or give the information the most quickly possible. The layered architecture reacts very slowly to the occurring events in a dynamic environment like MANETs and hence the produced lateness can make some notifications simply insignificant and cause the network to waste considerable resources. TCP for example is a protocol which first, must be rethought for wireless environments because it was built on an assumption which is no more valid and second, needs to be notified timely about losses or congestion which require new architectural adjustments. Another factor which pushes designers to break the layered architecture is the need to optimize some network's parameters, like the consumed energy or the produced delay, which are influenced by multiple layers. In such cases, designers often prefer to couple the design of some layers to get optimized parameters. At last, it is important to recall that some layers have inherent dependencies. It is the case of the MAC layer and the network layer or the MAC layer and the physical layers where additional interactions or design coupling are more appreciated like to choose the transmission mode or rate in the physical layer with conformance of the frames nature or the error correction used.

6.4. The cost of cross–layer optimizations

Cross–layering is a perspective design principle which aims to adapt existing protocols to the wireless case and to achieve performance improvements in wireless networks via the exploitation of the new opportunities offered by wireless networks and also by adding

new functions to face the new issues created by the wireless link. Adopting such approach can probably allow attaining the aimed performance levels in wireless networks and especially in ad hoc networks. But, this comes at a price. Proceeding with Cross–layer design implies that we abandon the advantages of the layered architecture and almost certainly face new challenges.

6.4.1. Design, implementation and modeling complexity

The layering principle has been long identified as a way to provide simplicity to the designers of protocols. The networking service is divided into a set of hierarchical layers where each layer offers services to adjacent upper layers and requires functionalities from adjacent lower ones. The designer in this case is only required to have the full specification of the module he is working on an only a global knowledge on system. This modularity is the main factor behind the exponential evolution and proliferation of many technologies like for the internet or computers because it eases both design and upgrading. It also accelerates the systems development process by allowing parallel efforts without ignoring the need to understand the entire architecture even if the designer is working only on a part of it. At the other side, resorting to cross–layer design implies coupling design of many modules or defining new bigger building blocks. This makes the task of designers heavier and hence slow down the evolution of the related systems. Cross–layer design also increases the implementation complexity as it does for design since more complex designs provoke also complex implementations. The implementation issues are also aggravated by the non standardized interactions which make the code more difficult to understand and maintain. Modeling and studying the system also becomes more difficult with cross–layer design. This in fact is a big obstacle that prevents the proliferation of cross–layer designed systems.

6.4.2. Interoperability with existing systems and coexistence of multiple CLDs

The consistency of the Internet shows clearly how the architecture is very important in ensuring the interoperability within any system. Cross–layer design proposals are diversified and usually create additional dependencies between layers to make the system more flexible and reactive. This is achieved through adding new interactions or design coupling of many components. However, such non controlled and non standardized

interactions can create serious problems in term of interoperability with existing systems or with other cross–layer designs even in the same system. In the layered architecture the interactions are controlled and the designer will never worry about how the other components will be designed since they fill their requirements. In contrast, using some cross–layer designs together can produce unintended interactions which can drop the system performance.

6.4.3. The system longevity

The longevity of the system under design is of economic and strategic importance. The adopted architecture is central for longevity of the system. In the layered architecture individual modules can be upgraded without necessitating a complete system redesign. This ease in upgrading the system drives to its longevity. On the other hand, using cross–layer design can lead to important immediate performance gain. However, we must consider how long we can benefit from it since a powerful system with short term benefits may be less profitable comparing to a system which offers lower performance but high longevity.

These factors show that it is not sufficient to get better performance results when resorting to cross–layer design. That does not mean that cross–layer design can't be a good choice in such situations however a trade off between performance and architecture must be considered. Claiming that any cross–layer design is effective for a configuration must be founded by proofing the performance and stability in the supposed environment. The cost and the longevity of the system also must be considered.

6.5. Cross–layer design proposals: a taxonomy

The key idea of cross–layer design is to maintain the original functionalities associated to the layers where allowing additional interactions, information sharing and coupling design of multiples layers. However this concept is very large and several cross–layering approaches have been proposed in the literature. There is no agreement on what a cross–layer design should respect and any proposal that violate the known layered architecture is classified as a cross–layer design. The existing cross–layer designs can be classified according to multiple criterions like the involved layers, the amplitude of the violation or the kind of this violation. In [87] the authors distinguish only two classes of

cross–layering namely, the weak cross–layering which only enables "non–adjacent" interaction among entities at different layers of the protocol stack and the strong cross–layering which also enables joint design of the components implemented within any entity at any level of the protocol stack. In [88] the authors propose a classification based on the kind of violation of the reference architecture. The authors distinguish six classes, the designs with added feedbacks from lower to upper layers, those from upper to lower layers and bidirectional interactions plus merging layers, design coupling and vertical calibration. Our classification is based on the same criterion and offers a clear view of what has been done in this area.

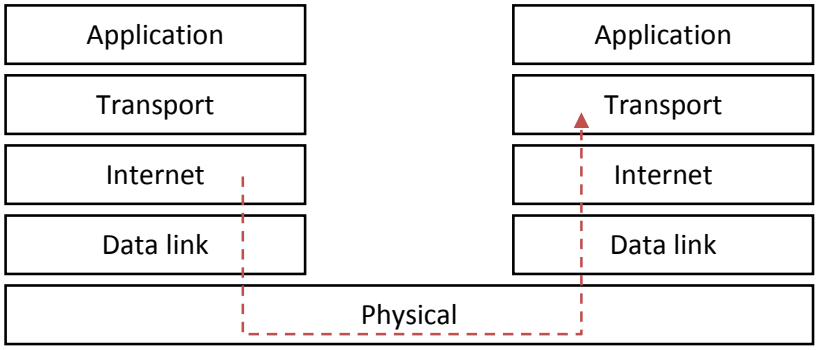


Figure 6-2: Notifications: the ECN.

6.5.1. Cross–layering based on added notifications

A notification is a message that carries explicit cross–layer information. One form of cross–layering is to add some notifications from one layer to the other in the same stack or in a remote node. Explicit congestion notification (ECN) [85] is an example where TCP is to be notified by remote network layers about the occurrence of congestion in the network (fig. 6.2).

6.5.2. Cross–layering based on added interactions

Additional full interactions are added between non adjacent layers (fig 6.3). The goal is to create shortcuts to deliver information quickly especially in dynamic environments like ad hoc mobile networks. At the extremity of this approach the original architecture can be completely changed and get dropped its hierarchy. A new abstraction can be then considered

where there is no more upper and higher layer but only neighboring layers. Some layers to get some information and other ones to get some services.

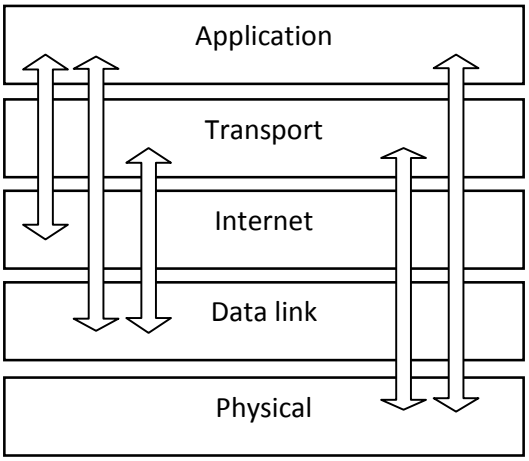


Figure 6-3: additional interactions between layers for cross layer design.

6.5.3. Merging adjacent layers

Boundaries between some layers are inherently thin therefore a designer may envisage merging some adjacent layers together to create a more powerful super–layer. This is the case for MAC and Physical layer in wireless networks where many proposals suggest intense interactions between them ([80] for example). Even if such proposals do not really attempt to merge layers, the intensity of the suggested interactions tends to blur the boundaries between them.

6.5.4. Design coupling

Coupling between multiple layers at design time without creating any new interfaces is also a cross–layer design. In this approach no additional interaction is needed and the layered architecture seems to be respected. However, the added behaviors at the corresponding layers form virtual interactions. Indeed, the principle here is that if any layer has specific abilities (like specific transmission or reception techniques, for example [81]) then the other layers can be designed with the capabilities of that layer in mind.

6.5.5. Cross–layering by adding an information sharing mechanism

In this case the designer offers a mechanism to allow layers to share information between them. This can be done via a shared database like in MobileMan [82]. This approach allows a full compatibility with the layered architecture while offering access to important network state information to the conscious protocols. One can also propose an extra vertical layer which interact with all the stack layers and maintain the state information (fig. 6.4).

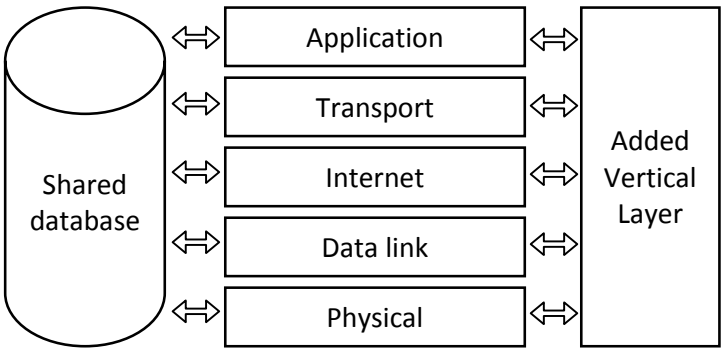


Figure 6-4: Cross–layer information sharing.

6.6. Examples involving cross layer

6.6.1. Cross–layer congestion control

Congestion occurs when the amount of data circulating in the network exceeds the capacity of it routers at some burdened points. This means that at some routers of the network the used buffer space has reached its maximal capacity. This situation causes data to be lost which decreases the network reliability. TCP is the main transport layer protocol and its purpose is to offer reliable end–to–end packets delivery. To achieve that reliability TCP is also required to control the congestion in the network. At the origin TCP attributes any packet loss to congestion in the network and reduces the transmission rate. This is because it was designed for wired networks where the packet loss is almost due to congestion. However in the wireless networks the probability packet loss due to bit errors is important. And with the traditional version of TCP the packet loss due to bit error would be misinterpreted as congestion in the network and cause the sender to decrease its rate which decreases the

throughput. Many solutions have been proposed to cope with this problem in wireless networks. A few of them are specific to ad hoc mobile networks, our subject in this document. We will focus on these ones and more specifically on the proposals which have a cross layer aspect. The first example we should give is the ECN (Explicit Congestion Notification) [85] added to TCP in wired networks. When congestion occurs at a router some packet will be inevitably dropped where some other packet will be properly routed. An ECN indicator bit is marked on the lucky packets and the receiver is made aware of the congestion. This information is explicitly sent to the source in order to decrease the transmission window. In other words the network layer in the congested router notified the transport layer of the receiver about the congestion. The second example is the TCP–Feedback or TCP–F [83]. It also utilizes the network layer feedback. If a router is unable to reach its next hop then it sends a Route Failure Notification to the TCP sender. When the source (i.e. TCP sender) receives the packet, it goes into persist state, in which it stops sending packets, invalidates its existing timers and freezes its TCP window, and state. Similarly, and when the route is reestablished the TCP sender is sent a Route Reestablishment Notification packet. On the reception of this packet TCP returns to the active mode with the stored parameters. ATCP (Ad hoc TCP) [84] is another proposal based on network layer feedback. It relies on the network to generate appropriate ICMP host unreachable messages and send them back to the source which cause TCP to enter persist state. ATCP inserts a thin layer between TCP and IP. It listens to the network state information by monitoring ECN (Explicit Congestion Notification) messages and ICMP messages, and then puts TCP at the sender into the appropriate state. These messages allow the TCP at the sender to distinguish three states, the route failure, the packet loss and the network congestion which helps it to perform better. The last example we'll give here relies on full cross–layering rather than feedbacks. We mean LLE–TCP (Link Layer ARQ Exploitation) [86].

The Link Layer ARQ Exploitation TCP (LLE–TCP) is proposed to exploit the information of the link layer ARQ scheme for a more efficient acknowledgement of TCP packet delivery [86]. The key idea is that when a TCP packet is successfully delivered at the link level, the TCP ACK for the transport layer is automatically generated locally at the sender side. In order to support this functionality a new software entity called the ARQ agent is introduced. The approach is available for both centralized and ad hoc mode. In centralized

architectures only one hop separates the node from the base station and the ACK created by the ARQ agent when the transmission is successful will not create troubles to it. However ad hoc networks are multi hop and the sender ARQ agent can face significant troubles if it acknowledges TCP packets when they successfully pass the first hop to the destination. Indeed, LLE–TCP is adapted to the ad hoc mode as follow. LLE–TCP ARQ agents operate at the last hop router (LHR) and the receiver (Fig. 6.6). LHR generates TCP ACKs relying on link layer acknowledgements on the last hop. An additional LLE–TCP Congestion Control (LLE–TCP CC) module is inserted in the protocol stack of sender node. Its role is to move the position of congestion control from TCP layer to itself in a lower level in the same protocol stack. This module can monitor the TCP transmission rate according to the received acknowledgments and the available buffer space.

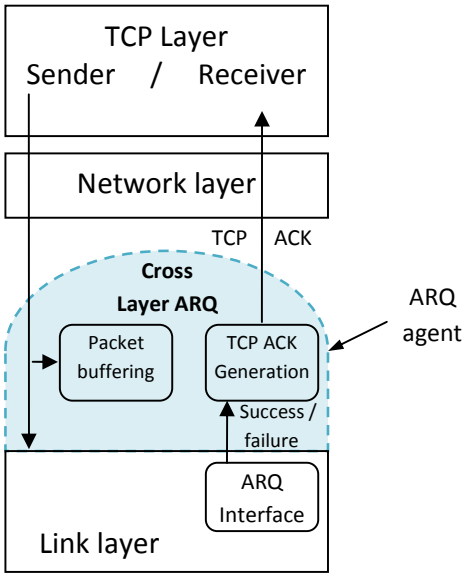


Figure 6-5: LLE–TCP, the ARQ agent position [86].

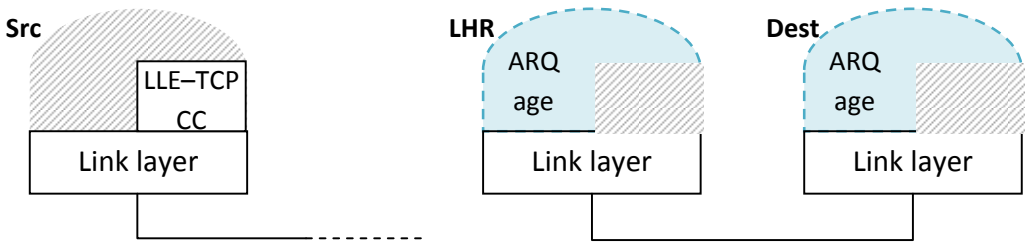


Figure 6-6: LLE–TCP in multi–hop networks.

6.6.2. MOBILEMAN

MobileMan [82] is a cross–layer architecture for ad hoc mobile networks that introduces inside the layered architecture the possibility of sharing any parameter between any pair of layers. It proposes a common database that can be accessed by all the layers. Thereby, It keeps the advantages of the layered architecture where allowing cross–layering via network status information sharing (fig. 6.7). This approach saves important effort gains because the shared parameters are to be calculated by the closer layer and not by the protocol which need them. Such parameters can be then exploited by multiple components in the network stack while maintained by only one component.

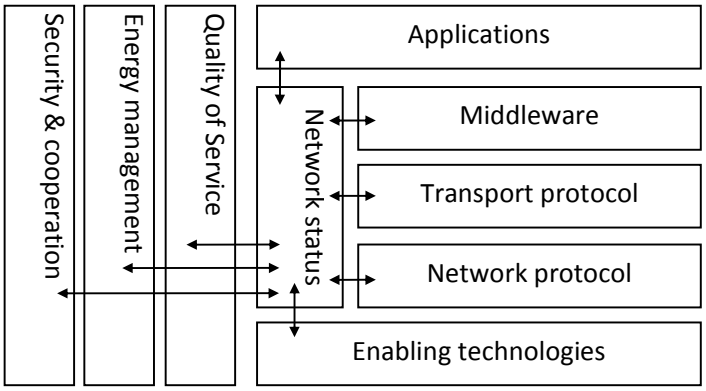


Figure 6-7: MobileMan reference architecture.

However, an important problem with this approach is when implementing it. Indeed, it is not evident to envisage a shared buffer between the stack layers since layers are implemented at different levels. The physical layer is implemented in the network interface firmware, the link layer can be implemented with the physical layer or in the peripheral driver, and the rest is distributed between the operating system kernel, its services and the applications.

Energy management, security and cooperation are cross–layered by nature, as seen in Figure 6.7. The core component in this reference architecture is the Network status repository. Whenever a protocol in the stack collects information, it will make it available for every other protocol. Thereby, avoiding duplicating efforts to collect information as sensed above.

6.6.3. CrossTalk

The CrossTalk [89] architecture for ad hoc networks is another cross layer architecture. It is similar to MobileMan. The difference with CrossTalk is that it adds the global view to the shared state while MobileMan only consider local state. The global view is constructed by the CrossTalk' data dissemination process. Data packets carry the source local state information to share it with other nodes and form the global state. Having such a global view, a node can use global information for local decision processes in conjunction with a local. The main drawback of such approach is the produced communication overhead. Local information to be shared is piggybacked onto outgoing data packets and no particular message is created.

6.7. Performance of cross–layer designs: a bad example

The layered architecture allows limited interactions between the layers which offer design and stability verification simplicity. A cross layered approach can envisage to add some interaction which leads to additional dependencies between layers. If only one cross–layer design is adopted then the designer must verify that the created independencies do not generate any undesirable effect with the original architecture. But when putting multiple cross–layer designs together new dependencies can appear. Those ones haven't been imagined at the design time and can totally drop the system performance. Such interactions need to be studied using dependency graphs to investigate formed loops and study their effects on the system performance. Following we will present an example which has been studied in [43,113].

6.7.1. Rate adaptive MAC with minimum hop routing

The key idea in Rate adaptive MAC [91] is to use higher rates when the channel conditions are better. It is a variant of the IEEE 802.11 MAC protocol. A predefined set of rates are available at the physical layer and each packet can be sent at a different rate. The broadcast packets are transmitted at the lowest data rate. It is the case for RTS and CTS. The receiver estimates the channel quality with a node by measuring the received signal strength of the RTS packet. This estimation will be used to decide about the rate at which data will be

sent. This rate is communicated to the sender in the CTS packet. This idea is reasonable to reduce the loss ratio. However, using such an approach with a minimum hop routing protocol can lead to unintended consequences. Indeed, using the path with the optimal distance means that the hops forming this path are larger and the signal strength is lower. This drive rate adaptive MAC to use lower rates on this path (fig. 6.8). Thus the throughput is reduced.

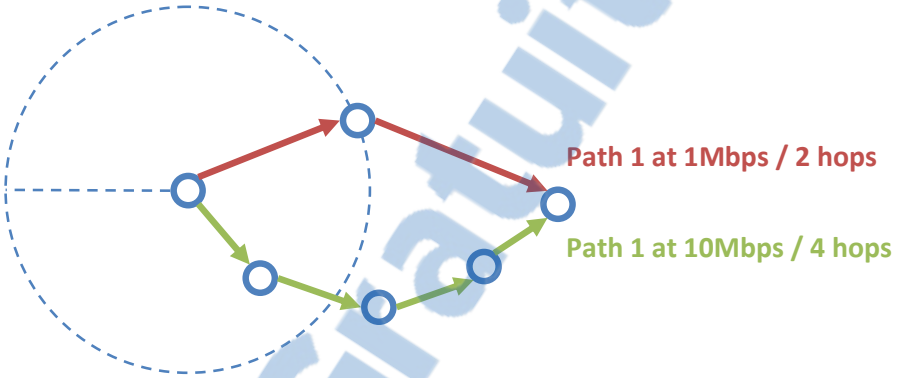


Figure 6-8: Rate adaptive MAC + Minimum hop routing.

6.8. Conclusion

Cross–layer design for mobile ad hoc performance improvement is an active research board. It is supposed to face the problems which came with the use of the wireless channel. It is also supposed to exploit the new offered opportunities. At the end of this chapter we come to some conclusions. It is clear that resorting to cross–layer design is currently the only available choice to answer today application needs. This approach allows better exploitation of available resources. Common network parameters can be optimized through CLD. However, this approach also has many drawbacks. It increases the complexity of the systems which slow down the evolution, threatens their longevity and increases their costs. More work is needed in this area. Perhaps proposing new approaches to specific systems can find success but the more important is to define some points of reference to organize the researchers' efforts. This may be done firstly by defining clearly the possible gains of cross–layer design. Secondly, it is indispensable to put standardization on how cross–layer design should be achieved because the main problem with cross–layer design is the lack of standardization. This standardization should consider the main problems like information to

be shared, the interaction to be added and the implementation choices. The standardization will lead to alleviate the complexity met by the designers. Otherwise, multiple works in all the directions will never mark an advance in this area and such unorganized efforts will get wasted for modest progress. Another available choice is to completely rethink the architecture to support and exploit today's technologies. Of course this choice is very expensive and can't be envisaged for the short–term future.

Knowledge without action is futile.
Arabic

Chapter 7

7.EFORTS
Every node Feedback for Optimizing Real time Traffic Support

Abstract

Current IP networks do not offer strict QoS guarantees to multimedia applications. This is more obvious when speaking about ad hoc wireless networks because of the limitations of these networks. In this chapter we present the main aspects of VoIP as a commonly used application and we then describe a cross layer optimization for better support of these applications.

7.1. Introduction

Voice over IP (VoIP) also called IP telephony or voice packet enables voice conversations over IP networks like the Internet. This technology allows sending voice on IP data networks rather than by circuit–based architectures of public switched telephony networks (PSTN). It is having a big progress this last decade and is currently among the hottest research topics. In the Internet and in all the wired networks VoIP has become a common application because of the low cost it entail comparing to public telecommunication

networks and the flexibility it adds. Using VoIP also goes well with the current tendency to integrate data networks and multimedia services networks. People now hope to get all the services on the Internet which they can access using different devices. However, quality stays a key issue of VoIP. This, because it requires special real–time support which it is not the purpose of IP networks. PSTN establish dedicated channels to transport voice packets so the needed resources are allocated to offer a determined service quality. At the opposite, IP networks are best effort at the origin and traffics therein are more dynamic and unpredictable so no absolute reservation is possible. In ad hoc mobile networks, VoIP is the most important application. If we take the main applications of MANETs we find the tactical military networking and emergencies operations where conversations are the most needed service. At the same time we know that ad hoc networks suffer from the lack of resources and the instability. This make offering VoIP service in such environment a very challenging problem. This pushes us to investigate a new track to support such application. This track consists of a cross–layer design.

7.2. Why VoIP?

As we said above IP telephony is having a big growth. This increasing interest is due to many advantages brought by this technology. Mainly the low cost comparing to telecommunication networks and voice integration with data networks which offer new and elegant functionalities (Call me link on your web site for example, in a forum or any internet application). VoIP is a good solution in many contexts but still have some limitations to replace telecommunication networks for now because the required QoS can't be guaranteed on the large scale Internet.

7.3. VoIP transmission

Voice transmission on IP networks is done in three steps: Digitization of the analog voice, coding to remove redundancy and packetizing it before it can be sent. These operations are inverted for the reception.

7.3.1. Digitization

This operation consists of transforming the analog voice signal to a binary string. It is inherently a lossy process. Since the nature of the speech signal is analog, an analog to digital conversion is necessary. This conversion conceptually entails two different processes: sampling (continuous–time to discrete–time) and quantization (real–valued to discrete–valued). The bit rate (BR), number of bits per second, is obtained as the product of the sampling rate and the number of bits per sample.

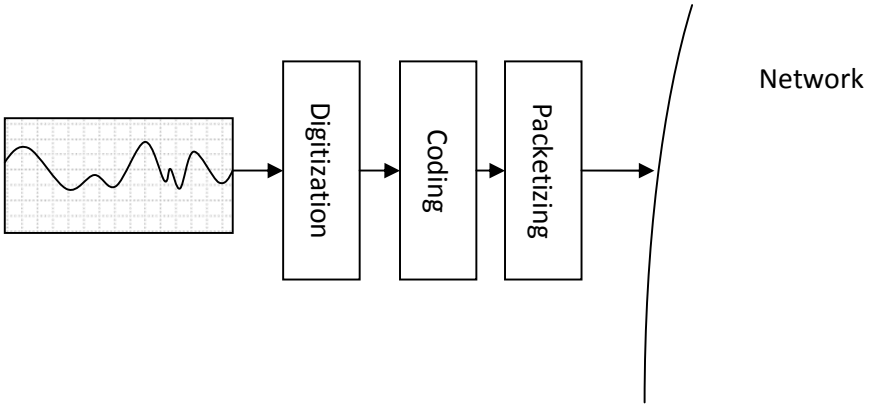


Figure 7-1: VoIP transmission.

7.3.2. Speech coding

Speech is a non stationary and redundant signal (varying statistical properties change over time). Speech coding mainly consists of removing redundancy. The non redundant part of speech is then encoded in an adequate manner.

Speech coding is required for efficient communications (VoIP for example) and digital storage of speech (Answering machines, Voice mails …). Speech Coding for IP and Wireless Is based on one of three types of coders, namely, waveform, parametric, and hybrid coders [93].

7.3.2.1. Waveform Coders

Waveform coders attempt to preserve the wave form of the original signal. The signal is coded and transmitted on the channel. They operate on a sample–by–sample basis, and consequently do not incur in any time delay (the algorithmic delay is one sampling period), and typically operate at medium to high bit rates (32 kb/s and above). Pulse Code Modulation

(PCM / G.711), Differential PCM, Adaptive DPCM (ADPCM / G.726) and Continuous Variable Slop Delta Modulation (CVSD) are the main coders of this family.

7.3.2.2. Parametric Coders

Parametric coders assume a simplified model of the speech signal. Thus, this model, irrespective of the waveform, keeps the perceptually relevant characteristics of the speech signal. Only parameters are coded and transmitted over the channel for reconstruction of the original signal. These coders work on a frame–by–frame basis. For every frame, the model parameters are estimated, quantized and transmitted. They generally operate at low bit rates [93]. The LPC vocoder (Linear Prediction Coder / voice coder) is main coder in this family. Two new coders of this class are the mixed excitation linear prediction (MELP) and waveform interpolation (WI) vocoders.

7.3.2.3. Hybrid Coders

Hybrid coders fill the gap between waveform and parametric coders. On the one hand, they assume a speech model; on the other, they try to preserve the original waveform. Hybrid coders work on a frame–by–frame basis and typically operate at medium bit rates (between 4 and 16 kb/s). These coders are also known as analysis–by–synthesis (ABS) coders. The main coders of this class are RELP (Residual Excited Linear Prediction) and CELP (Code Excited Linear Prediction).

7.3.2.4. Speech Coders used in Mobile Radio Systems

Standard	Service Type	Coder	Bit rate (kbps)
GSM (FR)	Cellular	RELP	13
IS–95	Cellular	CELP	1.2 – 2.4 – 4.8 – 9.6
CT2	Cordless	ADPCM	32

Table 7-1:Speech coders.

7.3.2.5. Coder Attributes

When designing a VoIP system, the choice of a speech coder is function of a number of network factors such as the expected delay and the available processing power, as well as the user requirement of service quality. The attributes of a speech coder include bit rate, complexity, delay, and voice quality.

a. Bit rate

The bit rates of the coders defined by the ITU range from the low 2.4 kbit/s coders to the 64 kbit/s wideband coders, such as the G.722 or the G.711 pulse–code modulated (PCM) coder. The rate of the coder determines the required channel bandwidth. Obviously, the higher the bit rate, the better the quality. Speech coders for wired IP are either waveform (16 to 64 kb/s) or hybrid (5.3 to 16 kb/s), while for wireless the selected coders are either hybrid (3.45 to 13 kb/s) or parametric (1.2 or 2.4 kb/s, generally). Traditionally, as required by communication networks, speech coding algorithms have been designed to provide a fixed bit rate (FBR) However, in the last several years, two types of non–FBRs have risen: variable bit rate (VBR) and adaptive multi–rate (AMR). These coders are of big importance in the context of wireless networks.

b. Delay

The delay of the coder is relevant issue in both IP and wireless. It is more critical for voice over IP (VoIP) since there are several subsystems that add other significant delays. The total delay of a coder includes the framing, look–ahead and processing delays. The average delay should be kept below 150ms. This delay can be relaxed to 300 ms for lowly interactive conversations. Above 300 ms, the conversation becomes unpractical. The algorithmic delay typically takes values between 15 and 40ms.

c. Complexity

Complexity is another important factor. Maintaining the quality, the bit rate can be reduced by increasing the complexity. In the wired IP environment, complexity is less critical, since current PCs are powerful enough to run any standard coder in real time. In wireless networks it is more relevant because complexity means power consumption and, consequently, reduction of battery life. Therefore it is always desirable to have efficient algorithms that do not use up a large percentage of the available processing power.

A speech coder must offer additional possibilities like flexible adaptation to network condition changes. It should be designed to gracefully degrade when one or even several consecutive entire frames are lost (one packet could contain one or more frames, and packet loss typically occurs in bursts). Moreover, wireless communications systems are prone to transmission errors.

Attribute	G.723.1	G.729	G.729a
Bit rate	6.4kbit/s 5.3kbit/s	8 kbit/s	8 kbit/s
Frame size	30 ms	10 ms	10 ms
Look ahead	7.5 ms	5 ms	5 ms
Total delay	67.5 ms	25 ms	25 ms
Complexity RAM	16 MIPS 2.2 kwords	20 MIPS 3 kwords	10 MIPS 2 kwords

Table 7-2: Summary of Attributes for 3 Commonly Used VoIP Coders [95].

7.3.2.6. Silence Suppression

Silence suppression removes the periods of silence that occur naturally within a voice conversation. The main cause of silence is when one peer is listening, but other shorter periods of silence occur between sentences, phrases, words. Silence accounts for nearly 60 percent of the bits sent during a two–way 64–kb/s PCM voice conversation [96]. This ambient background noise must not be sent, only the speaker's voice. The trick is to reliably detect when the speaker's voice level has risen high enough above the background noise to determine that the speaker has actually begun to talk. This is done via a technique called voice activation detection (VAD) at the speaker.

Parameter	Rate (%)
Talk–spurt	38.53
Pause	61.47
Double talk	66.59
Mutual silence	22.48

Table 7-3: Temporal parameters in conversational speech (average for Eng.) [96].

7.4. Quality of VoIP

Good QoS means providing adequate service to the end users. IP data networks were not originally conceived to transport real time traffic but were only supposed to carry best–effort traffic. Today and because of the big growth and success of the Internet has known, people start to think about integrating new and may be all the services on this IP network. The VoIP concept came out as a result this tendency. However, VoIP traffic requires a QoS level and imposes additional constraints on the underlying network. The following is a

presentation of the issues that may make IP networks unsuitable or at least challenging (a priori) for voice transport.

7.4.1. End–to–end delay

End–to–end delay consists of the time a packet takes to reach the receiver. Real–time voice communications are sensitive to delay. A threshold is determined (an average of 150 ms for common telephony conversations). Between 150 ms and 300 ms users will feel slight hesitation in their partner's response. The conversation becomes cold. Beyond 300ms the delay is obvious to users, and they start to back off to prevent the interruptions. ITU–T standards state that end–to–end delays of up to 150 ms are fully acceptable, delays greater than 400 ms are unacceptable, and values in–between are acceptable but noticeable by users. Acceptable delays are not easy to achieve because an IP packet network does not reserve resources for the duration of a call. Each packet has to compete for resources at every intermediate node it traverses, so its delay is affected by the number of hops and by traffic patterns at each node. Once the delay is in the system it cannot be removed then only prevention is allowed.

Controlling packet delays is the main technological problem in most IP networks. Delays in packet networks result from framing, jitter control, processing and the variation in network conditions.

7.4.1.1. Framing

Framing delay consists of the time to collect and frame the samples. The value is function of the coders used (e.g. 10 ms for G729a; 30 ms for G.723). The coder compress fixed size chunks of linear samples, rather than sample per sample (Figure 7.2). Therefore, the audio data stream needs to be accumulated until it reaches frame size, before being processed by the coder. This sample accumulation takes time and therefore increases end–to–end delay. In addition to that, some coders need to know more samples than those already contained in the frame they will be coding (this is called look–ahead). Therefore, in principle, the codec chosen should have a short frame length in order to reduce delays on the network. However, many other factors should be taken in consideration. Primarily, coders with larger frame sizes tend to be more efficient, and have better compression rates (the more you know about

something the easier it is to model and treat it efficiently). Another factor is that each frame is not transmitted 'as is' through the network: a lot of overhead is added by the transport protocols themselves for each packet transmitted through the network. If only one voice frame is transmitted in a packet the overhead will be greater than the useful data therefore most implementations choose to transmit multiple frames in each packet; this is called 'bundling'. Of course bundling adds more accumulation delay.

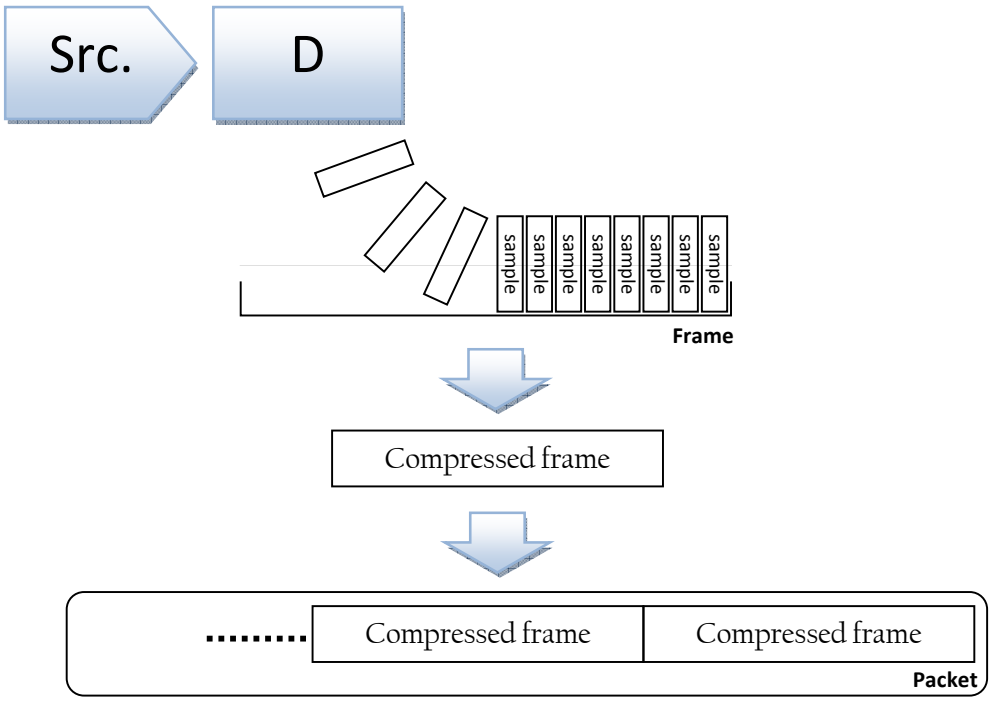


Figure 7-2: Framing.

7.4.1.2. Jitter control

Jitter buffers, used to control jitter, also introduce delay. If the jitter is important frames will be delayed to the max possible value.

7.4.1.3. Processing delay

This delay depends on the user Operating system and equipment and the speech coder design and implementation. This delay is usually acceptable or small but not optimal. Most VoIP applications are regular programs running on top of an operating system, such as Linux or Windows. These OS are not real time operating systems and may add additional delays. They access sound peripherals through an API and the network through the socket

API. As you speak the sound card samples the microphone signals and accumulates samples in a memory buffer. When a buffer is full the sound card tells the operating system, using an interrupt, that it can retrieve the buffer, and stores the next samples in a new buffer, etc. Interrupts stop the current task and launch the interrupt handler which will return the buffer to the demanding application which uses the socket API to send after suitable encoding.

7.4.1.4. Network delay

It is the biggest challenge facing VoIP because it has higher variation and levels comparing to processing or framing delays. The accumulation of delays except the network delay is generally under 60ms. The network delay consists of the time spent by frames in routing from the source to the destination. It includes queuing, transmission and propagation time.

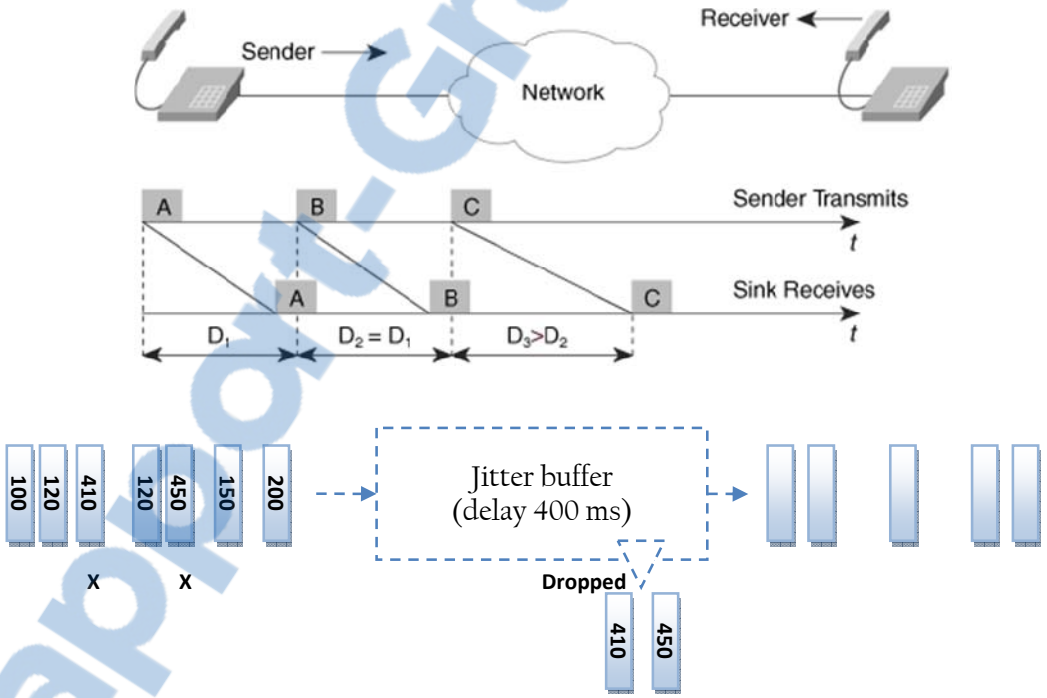


Figure 7-3: Jitter control.

7.4.2. Jitter

Jitter is the variance in the E2E (end to end) delay. It is due to the fact that the IP networks do not establish a dedicated channel to each flow. IP networks proceeds on a packet–by–packet basis or with soft reservations in best cases which results in a different

faced situation for each packet and hence produces variable delays. Even if intermediate routing of traffic provides priority to voice traffic, there is no guarantee that consecutive packets arrive in order at the destination.

To cop with this phenomena VoIP receivers use jitter buffers to convert the variation to a delay or a loss. A jitter buffer is simply a FIFO (First–In, First Out) memory cache that collects the packets as they arrive, forwarding them to the codec. This produces an additional delay. While a jitter buffer can successfully mask moderate jitter problems, severe jitter results in packet loss. The size of jitter buffer can be dynamic to allow an optimal profit of the network conditions.

7.4.3. Loss

Because IP networks treat voice and data with the same manner, voice packets will be dropped under severe traffic loads. In addition to that VoIP packets can be dropped if they accumulate a non tolerated delay. Furthermore, it does not worth to resend lost voice packets. This makes Voice flows more suffering from loss.

Codecs should be robust such they can handle some packet loss. However a loss higher than **5%** will be inevitably annoying. The amount of packet loss a codec can handle before voice performance goes down depends on the used algorithmic. The packet delivery ratio is given by:

$$pdr = \frac{Amount\ of\ received\ data}{amount\ of\ sent\ data}$$

$$loss\ ratio = 1 - pdr$$

A voice packet can contain up to 40 ms of speech information. Therefore, loss has a big impact on VoIP QoS. The amounts of loss that can be tolerated depend mainly on the vocoder. Even a 1% loss can significantly degrade the user experience with some voice coders. The higher is the compression the more significant is the degradation in voice quality following packets loss.

In ad hoc mobile networks the loss ratio is more important because of the error prone nature of the wireless link and the unpredictably changing routes. These same conditions also

extremely affect the network delay. Retransmissions and re-routing can produce non tolerable delays. This makes it necessary to consider these two parameters when designing a solution to VoIP in such environment.

7.4.4. Voice quality measurement

Measures of quality tend to be subjective in communications systems, among the subjective metrics we find the Mean Opinion Score (MOS) defined in [98]. A MOS score can range from 1 (bad) to 5 (excellent. The Pulse Code Modulation (PCM) algorithm (ITU-T G.711) has a MOS score of 4.4. Other objective models such as the E-Model [100] attempts to predict QoS scores using more objective factors. The most popular objective measurements are Perceptual Evaluation of Speech Quality (PESQ) [107] and E-model [100]. PESQ requires the original speech signal with the degraded one to perform the quality evaluation while E-model is parameter-based and does not require the original speech signal.

The E-Model is a computational model, standardized by ITU-T in [100]. Another variant for VoIP has been published in [110], namely, Packet-E-Model. It tries to predict the subjective speech quality of packetized voice. The primary output from the E-Model is the "Rating Factor" R, and R can be further transformed to give estimates of customer opinion by mapping it to the MOS scale. A comparison of E-Model Rating Values (R) and MOS scores is shown in the table 7.4. Regarding the use of the E-model for speech applications, the effect of delay is shown in the graph of E-Model Rating, R, versus delay (figure 7.4).

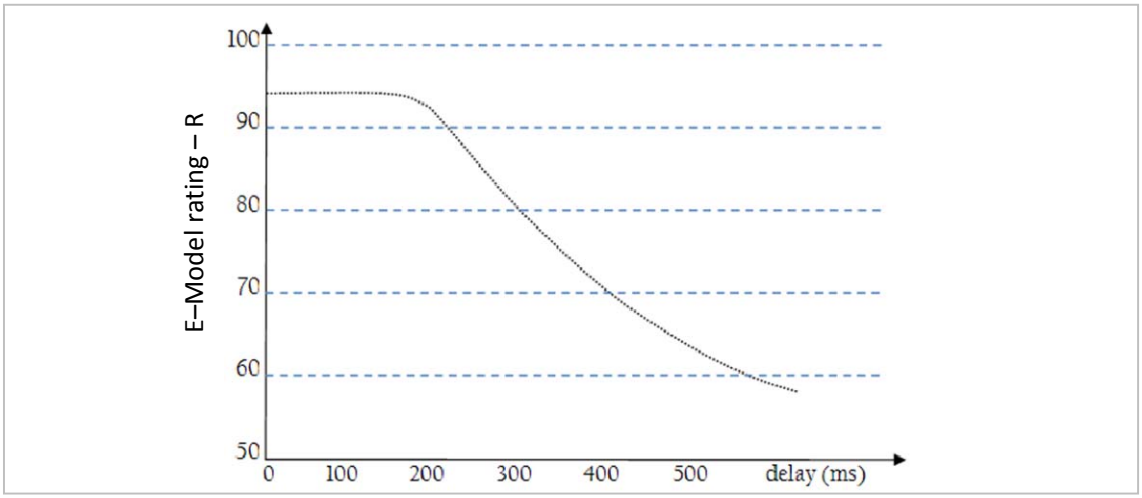


Figure 7-4: E2E delay impact on Voice quality [94].

R–value range	quality category	User satisfaction	MOS
90 ≤ R ‹ 100	Best	Very satisfied	4.3+
80 ≤ R ‹ 90	High	Satisfied	4.0 – 4.3
70 ≤ R ‹ 80	Medium	Some users dissatisfied	3.6 – 4.0
60 ≤ R ‹ 70	Low	Many users dissatisfied	3.1 – 3.6
50 ≤ R ‹ 60	Poor	Nearly all users dissatisfied	2.6 – 3.1
0 ≤ R ‹ 50	Not Recommended	–	1.0 – 2.6

Table 7-4: Speech transmission quality [99].

7.5. Real time protocols, RTP & RTCP

Real–time transport protocol (RTP) provides end–to–end delivery services for data with real–time characteristics. Those services include payload type identification, sequence numbering, timestamping and delivery monitoring [106]. RTP runs on top of UDP to make use of its multiplexing and checksum services however technically the RTP protocol is implemented in the application layer; both protocols contribute parts of the transport protocol functionality. But, RTP may be used with other transport protocols. Connection support makes TCP a natural choice for VoIP, but TCP is a reliable protocol which resend missing segments. This is useless for real–time VoIP therefore UDP is the protocol used for VoIP. However, UDP connectionless nature makes it more difficult to map VoIP connections onto the IP network. This explains why RTP is needed to support VoIP. RTP have similarities with TCP, but does not resend lost packets. This makes RTP suitable for all sorts of real–time applications, including VoIP. The main services RTP can add are loss detection, reception quality reporting, synchronization, payload and source identification and marking of significant events within the media stream. For VoIP, some have argued that RTP provides unnecessary features, and is heavyweight for highly compressed voice frames because of the added header in each packet. Two optional pieces of the RTP framework aims to alleviate this concern: header compression and multiplexing.

Header compression is a means by which the overhead of the RTP and UDP/IP headers can be reduced. It is used on bandwidth–constrained links and can reduce the 40–byte combination of RTP/UDP/IP headers to 2 bytes, at the expense of additional processing by the systems on the ends of the compressed link. Multiplexing is the means by which multiple related RTP sessions are combined into one. Once again, the motivation is to reduce overheads, except this time the procedure operates end–to–end.

The RTP control protocol (RTCP) provides reception quality feedback, participant identification, and synchronization between media streams. RTCP runs alongside RTP and provides periodic reporting of this information. Although data packets are typically sent every few milliseconds, the control protocol operates on the scale of seconds. The information sent in RTCP is necessary for synchronization between media streams—for example, for lip synchronization between audio and video—and can be useful for adapting the transmission according to reception quality feedback, and for identifying the participants.

7.6. Signaling protocols

Signaling is a key function in the telecommunications networks. Telephone calls are set up and terminated through signaling. Signaling also defines the desired service for the user, such as point–to–point calls, multipoint conferencing, text, voice, or video. Significant efforts were undertaken in past decades to develop the signaling protocols in use in public switched telephone network (PSTN). Similar efforts are now being undertaken to define voice over IP (VoIP) signaling.

The ITU–T Recommendation H.323 [104] and the session initiation protocol (SIP) [109] from Internet Engineering Task Force (IETF) are the two major emerging standards in the industry for VoIP. SIP has been adopted by practically all public VoIP service providers for wired and wireless communications. The discussions about SIP versus H.323 standardized by the ITU–T are over as well.

7.6.1. SIP

[111] "SIP is an application–layer control protocol that can establish, modify, and terminate multimedia sessions such as Internet telephony calls. SIP can also invite participants to already existing sessions, such as multicast conferences. Media can be added to (and removed from) an existing session".

SIP is a peer–to–peer protocol deployed for signaling between SIP endpoints (user agents and media gateway controllers, MGC). It supports user mobility through proxy servers and redirecting requests to the user's currently registered location. The SIP specifications are provided in RFC2543 of IETF.

In the next sections we target cross–layering as a possible solution, presenting a novel cross–layer approach, we called **EFORTS (Every node Feedback for Optimizing Real time Traffic Support)**, where we aim to optimize VoIP support through the rationalization of the network resources utilization by involving application, MAC protocol and routing.

7.7. VoIP in ad hoc networks

7.7.1. New challenges

As we said before VoIP traffic is not well handled in the Internet because IP networks are best effort at the origin and do not achieve strict resources reservations. In ad hoc networks this problem is more apparent because of the nature of these networks. We can summarize the added challenges in four points:

+ The network conditions of MANETs are varying and depend on many factors. This makes the amount of the available resources constantly changing where VoIP applications require regular service during the communication session.

+ Multimedia applications commonly use UDP protocol to transport their data and as we know UDP is not equipped with any congestion control mechanism. This can severely degrade the network performance and waste its resources which are very limited and precious in the context of MANETs.

+ Ad hoc networks are wireless networks and suffer from wireless channel losses. The loss probability experienced by packet transmission is high comparing to wired networks. This loss is due to the routes breaks, the interferences and the high bit error rate (BER) which varies from 10^{-8} to 10^{-6} for wired links and from 10^{-3} up to 10^{-1} for wireless channels. And as we know VoIP application has determined thresholds for the endurable loss ratios.

+ Using the wireless link also adds new security issues because this medium is easily accessible by intruders. In addition to that conventional encryption methods can add heavy loads on ad hoc nodes.

7.7.2. Convenient directions

To meet these new challenges new solutions must be investigated. Those ones must consider the nature and the characteristics of MANETs. Next we present the different directions that can be pursued.

7.7.3. Permissive error control

With the DCF of the 802.11 a frame is retransmitted if no acknowledgment is received immediately from the other node. A frame is acknowledged only it is correctly received. However some coders tolerate some amount of errors in their data and many retransmissions ad drops can be avoided if some other control algorithms are used instead of the common CRC.

7.7.4. Design of specific speech coders

The speech coding algorithm can be designed to tolerate the hard conditions of MANETs. It is always possible to use existing coders like G.711 when the network conditions allow it but the application must find alternative choices if the network capacity drop off.

7.7.4.1. Multiple description speech coding

An MD coder creates two (or more) coded bit streams from the same voice segment. If these two parts are correctly received the reconstructed speech will be of high quality otherwise the quality is still acceptable but lower. This technique can be combined with multipath routing in MANETs to improve VoIP support.

7.7.4.2. Scalable speech coding

This technique is similar to the previous one. The difference is that in this technique only one major bit stream is generated and a set of enhancement bit streams which have lower bit rates and priority. If the enhancements bit streams is also received in good timing they can be added to the major bit stream to increase the speech quality.

7.7.5. Renew network protocols design

As we said before the designers of MANET protocols has first tried to adapt existing internet solutions to the context of MANETs. This approach has shown its inability to face

the new challenges imposed by ad hoc networks. Then researchers have started to investigate new tracks to answer those challenges first with respect to the layered architecture and then lot of them has resorted to cross layer designs. The network protocols design stays the key element which has the biggest impact on the network performance. Therefore it must receive the highest attention and must consider the characteristics of the receiving context. In the fifth chapter we presented some designs that aim to support differentiated services in MANETs. The next section describes a new cross layer design which allows a more efficient use of the network resources and optimizes real time flows support in MANETs.

7.8. EFORTS (Every node Feedback for Optimizing Real time Traffic Support)

Along the previous parts of this document we have tried to show up the main characteristics of ad hoc networks, their two contradictory faces: big limitations and enormous promises. We presented with a critical view the main achieved works aiming to bring this class of networks into a convenient level. And also the works that aim to allow QoS support in this networks. We explained the reasons that pushed researchers to explore out of frontiers grounds trough cross layer design and we gave our view on how a cross layer design should be made. Above we presented VoIP as a real time application to which we aim to offer suitable underlying conditions through cross layer design. We come up to the description of our contribution.

7.9. Founding our proposal

EFORTS aims to optimize real time traffic support in ad hoc networks. The aimed metrics are loss and delay. It should allow better use of the network resources without adding big loads on that network. We founded our proposal on three objective principles:

7.9.1. 1st Ad Hoc networks are poor environments

As we explained in the first chapter, ad hoc networks are autonomous networks which are expected to operate in hard and least equipped environments. They use the wireless links to communicate which are shared and error prone mediums. Therefore ad hoc

networks are assumed to be poor environments where the networking resources are very limited and precious.

When designing a solution to VoIP support in ad hoc networks, the issue pointed above must be considered. Developing a high weight mechanism to offer strict QoS guarantees will produce a negative effect. Such solution will add a heavy load on this network's class while trying to offer something that probably does not exist at the origin. This will degrade the network performance and affect other networking applications in such a network without offering the intended services. Therefore we argue that the best solution in such context is to design a simple and lightweight mechanism that offers good enough QoS without overloading the environment by control overhead. The key idea is to devote the network resource to support VoIP traffic rather than investing big resources in attempting to fully control a complex and highly dynamic environment. We will simply accept that we do not have the required resources to get a full control on flows in the network and try to offer the required QoS with the lowest costs.

7.9.2. 2nd Ad Hoc networks are highly dynamic

In mobile ad hoc networks node are free to move with no restriction. This mobility makes the topology dynamic and unpredictable which affect routing tasks in the network and produce additional delays and losses. In addition to that, the wireless link is time varying because of the regular changes in the surrounding environment and also mobility. This makes the network resources at any corner of it unpredictably changing where multimedia application generates constant bit rate.

This pushes us to think that the application in these networks must behave differently. Trying to judge applications performances in ad hoc networks against their performances in conventional networks is a wrong move. Ad hoc network are very special environments that require unavoidably special use. The general approach that must be followed by such applications is to use adaptation in order to achieve their performance or QoS goals under such dynamic conditions. This can be done by reducing the produced bite rate or / and with changing encoding algorithm. An example is the use of error resilient encoding to enhance the robustness of compressed voice to packet loss. Such techniques can tune the amount of the redundant information according to the network state.

7.9.3. 3rd Ad Hoc networks offer new opportunities

The wireless link is broadcast by nature. This is the major functional difference with conventional networks. In ad hoc networks even if a node sends a packet to a specified node, all the nodes within its range transmission will receive that packet. This of course creates serious problems regarding interferences avoidance and data protection. Fortunately, this feature can be useful in many cases then we better should make use of it. Enabling promiscuous receive allows node to get free information. Such information allows nodes to be aware of the state and important changes in the network which help to better use its resources. This feature is very handy in many contexts like learning congested segments of the network. Such information can be used to improve signaling for resources reservation.

7.10. The proposed scheme (EFORTS)

EFORTS is an optimized ad hoc network for delay and loss sensitive applications such as VoIP. It is based on the principles described above and has four key elements:

- Useless packets must be discarded the earliest possible to save network resources.
- Application must be constantly notified by the lower layers about the important changes in the network to which it must react by adaptation. An example of such notification can be the RTCP feedback. However we think that such end to end feedback is not sufficient to solve any occurring problem. Therefore sharpest feedbacks are required.
- MAC layer protocol must be QoS aware and give priority to real time traffic taking in account the packet transmission deadline.
- QoS reports must be sent to notify the source or the destination application. But we also envisage that the nodes that can receive those reports can use the information therein to achieve harmless routing.

In VoIP traffics in IP networks a packet is never dropped before it reaches the destination even if it exceeds the maximal tolerated delay. This may do not affect performance in wired IP networks where the infrastructure is well–built and can support important loads. However,

in ad hoc networks bandwidth and power resources are precious and must be preserved as much as possible.

7.10.1. Packets deadline aware MAC

We propose to modify 802.11 / 802.11e MAC layer protocol by involving the packets deadline time of real time packets in the scheduling strategy. This deadline time will be used at the determination of the different parameters (backoff, CW) such as relative priority is given to packets with the closest deadlines (figure 7.5).

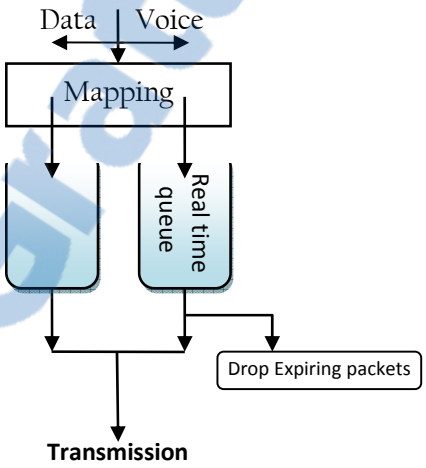


Figure 7-5: MAC real time scheduler.

If a real time packet expires before it gains the access to the medium it will be dropped. The statistics on expiring packets are achieved by a module in the MAC layer we called the "MAC Spy". It periodically reports them to a local agent in the network layer which we called the "Reporting Agent" which in turn reports them to the same agent ate the source of the related flow. Those reports are to be used to adapt application, improve routing and help local admission control (see figure 7.6). To decide if a packet has expired or not, no synchronization between nodes is required. We simply add a field to the packet which contains the remaining time for that packet to expire. This time is modified each time the packet is forwarded by subtracting the time spent in that node and the transmission time. The transmission time is very low comparing to the time spent in the node. It is about few µs and depends of the propagation delay and the transmission bit rate.

7.10.2. The MAC Spy

The MAC Spy is situated at the MAC layer. It constantly examines the real time traffic traversing that layer without altering its functioning. It keeps track of the arrival and departure time and expirations information to calculate three parameters. Those parameters consist of the amount of expiring packets, the mean of the generated delay at this node and its variation. This is done for all the traversing real time traffics.

If a real time packet expire in an intermediary node before it gain the access to the medium then this packet is no more useful. However, instead of being dropped simply the MAC Spy uses such information to determine the produced loss at that node.

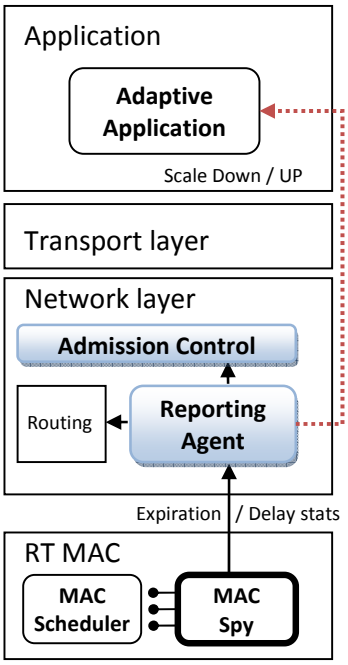


Figure 7-6: EFORTS design.

7.10.3. The reporting agent RA

The reporting agent is a software module who is responsible of keeping QoS parameters statistics and then using them to help supporting real time traffics. Those parameters are calculated by the MAC Spy module at the MAC layer.

The reporting agent at an intermediate node keeps the statistics about the lost packets in each flow and reports them to the same agent in the source of the flow (Figure 7.7–

b). The RA also keeps statistics on the generated delay at that node (mean, variation). It also notify admission control module about the QoS received by current admitted traffics.

A VoIP session can be unidirectional or bidirectional. The reporting agent at the receiver of the real time flow originates a state collector packet and sends it back periodically on the route the packets followed to reach it. The goal is to collect the statistics on each node and send them to the RA at the source.

Intermediate nodes on receiving the SC (state collector) simply add their ids and feedbacks to the source. Then they forward the SC to the preceding node (Figure 7.7–a / b).

On receiving the SC – which contains the feedback of the path's nodes – by the reporting agent at the source node, that RA then can order the application to scale down its bit rate if the loss ratio is higher than the tolerated threshold. For this to be possible the application must be able to operate at different bit rate levels. The RA before this step can order the routing protocol to investigate better routes by avoiding network bottlenecks reported by the SC.

The neighboring nodes can use promiscuous receive to learn free information on the state of some portions of the network. This information can by used by routing or admission control at the network layer to protect the ongoing real time sessions (Figure 7.7–b).

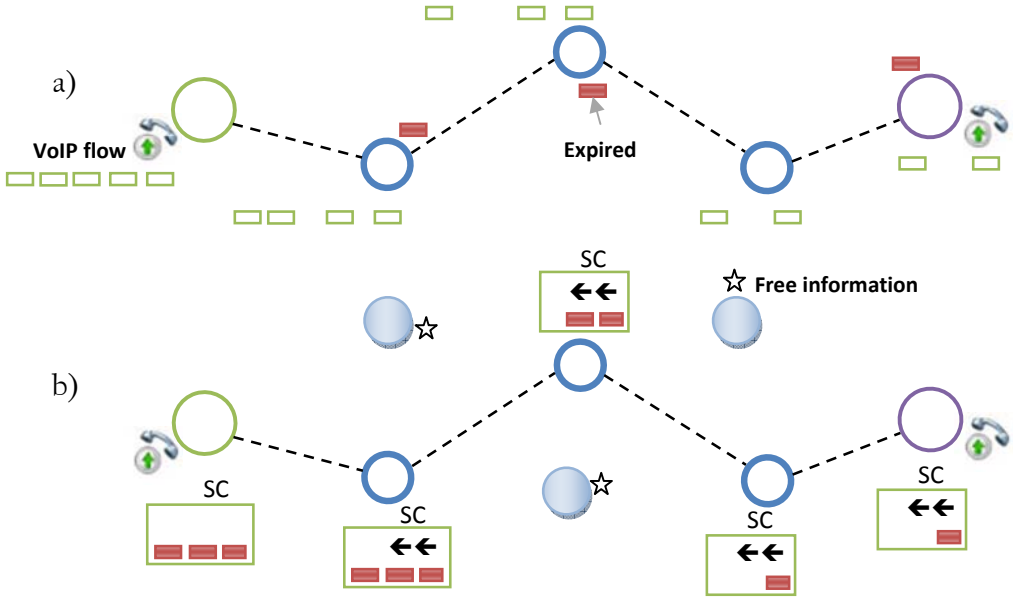


Figure 7-7: EFORTS State Collector.

7.10.4. The state collector SC

The state collector is a packet which is sent periodically from the receiver to the source through the path followed by the VoIP traffic. This packet contain two parts, the first part is allocated to the receiver feedback. It contains its identifier, the loss ratio, mean delay and delay jitter.

Receiver ID			
Loss ratio	Mean delay	Jitter	Number of entries

Table 7-5: The State collector – the head description.

The second part consists of a number of entries with one entry to each intermediate node. On receiving the SC each intermediate node appends its entry to the packet and increments the field Number of entries. The added values consist of the node identifier, the amount of expiring packets and the generated delay during the last period with the corresponding jitter.

Intermediate node ID			
Number of expiring packets during the last period	Mean generated delay in the last period	Generated delay jitter in the last period	

Table 7-6: The State collector – intermediate nodes entries.

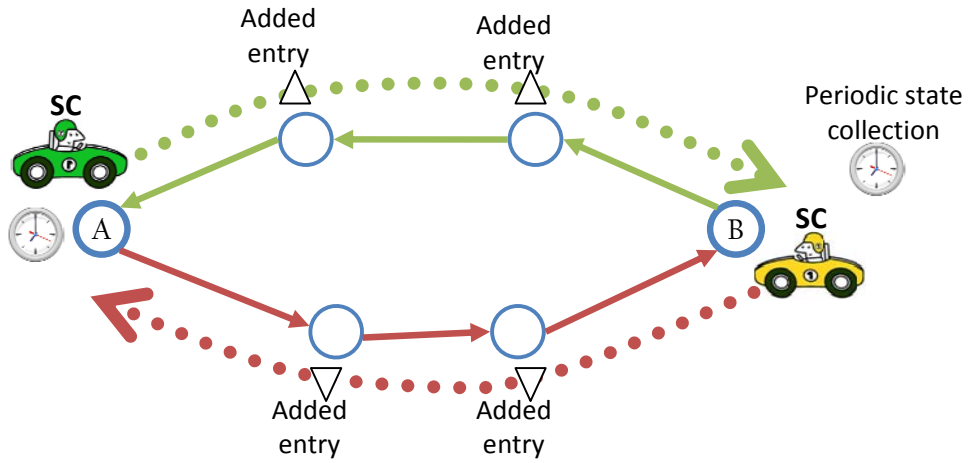


Figure 7-8: Periodic state collection.

7.10.5. RA interaction with routing

The information collected by the SC can be used to adjust routes computing. For example if a big loss ratio is reported by the destination's reporting agent the RA at the source node can notify the routing protocol that the current route does not satisfy the needed QoS. The routing protocol should then check for a new route. Therefore, the routing protocol would better have multiple cached routes all the time to avoid additional routes discovery delays (see TORA [17], OLSR [21] in chapter 3). Another example is when there are some bottlenecks on the route. In this case we can envisage asking routing protocol to avoid those nodes.

7.10.6. RA interaction with adaptive applications

As we said above, a key element in our proposal is to have adaptive application. The application will first attempt to run with the best QoS for the user. If the RAs don't complain then the flow has been supported by the network otherwise the application will receive a scale down order from the RA.

The RA scale down order causes application to scale down by one level. If the application was at the minimal supported level it will quit immediately. Therefore, sending RASD orders must be done after multiple and random number of attempts to find alternative routes at the network layer. The randomness here is imperative to avoid getting multiple applications shutting down at the same time which take the system from a congest state to an unloaded one. For example if a node experiences a congestion period all the VoIP traffics traversing that node will be affected and important delays and losses can be recorded. This information will reach the source nodes and can cause many applications to stop at the same time.

7.10.7. RA and admission control

The information available at the RAs in the intermediate nodes can be exploited in admission control. Such information show clearly the QoS offered to the current real time sessions thus an intermediate node can use this information to estimate if it can support additional sessions. Furthermore, neighboring nodes can use promiscuous listening to learn about QoS support at their range and hence can estimate if accepting any real time session can hurt existing sessions or not.

7.11. Simulations

In the previous section we presented our proposal and gave the thinking we followed to get into it. However that can't be sufficient to show the significance of the proposal or to validate it. The best way of course is to use experiments but this option is not accessible to us due to hardware limitations. Thus, for analysis and performance evaluation we have used simulation with ns2 (Network Simulator 2) [117]. Network simulation software is a costless solution and gives satisfactory results comparing to real systems experiments. It has become the most used validation approach by researchers for networking protocols. Ns2 is an event driven network simulator developed at VINT project (Virtual InterNetwork Testbed). VINT is a DARPA–funded research project whose aim is to build a network simulator that will allow the study of current and future network protocols. It is a collaborative project involving USC/ISI (University of Southern California / Information Sciences Institute), Xerox PARC, LBNL (Lawrence Berkley National Laboratory), and UC Berkeley (university of California Berkley). Ns2 is a free and open source object oriented program written in C++ / OTcl which is available under many OS platforms. It has an OTcl interpreter shell as the user interface that allows the input Tcl files to be executed. Network simulator 2 provides considerable support for simulation of TCP, routing algorithms, queuing algorithms, and multicast protocols over wired and wireless networks. However extending or modifying implemented modules stays a complicated task which requires a good understanding of the NS2 architecture but also good C++ and Tcl programming skills.

Even if this simulator is widely used in the networking research community we remarked that it is not well documented and this creates us additional difficulties.

7.11.1. VoIP support in unloaded MANETs

The first simulation we did consists of measuring the capacity of ad hoc networks to support VoIP traffics.

7.11.1.1. Simulation settings

In this simulation we used ns2.31 on Fedora Linux 8.0 running on a computer equipped with an Intel Pentium III (700 / 7x100) processor and 384 MB of SDRAM– PC100. We used the 802.11 for MAC/PHY layers and AODV, DSR and DSDV for routing but we will

present only the best recorded results which were those with DSR. With AODV performances were slightly lower comparing to DSR where DSDV was clearly inferior to them.

Ns2 implement the 1997 version of the IEEE 802.11 standard. The maximal data rate is estimated for 2Mbps and the basic data rate for 1Mbps. It uses the 900 MHz frequency band (the 914 MHz Lucent WaveLAN DSSS radio interface) which has been shifted to 2.4 GHz in the 1999 standard version. We achieved a set of tests on this implementation and then we modified it to match the 1999 version of the 802.11b standard which is currently the most manufactured and uses the IMS band. The parameters of the MAC and physical layers of the IEEE 802.11 implemented in ns2 are shown in the table 7.7 and the modifications we added to match 802.11b in open environments are presented in table 7.8.

Slot time	20 µs
SIFS	10 µs
DIFS	SIFS+2*Slots
PLCP preamble	144 bits
PLCP header	48 bits
Data rate	2 Mbps
PLCP data rate	1 Mbps
Basic data rate	1 Mbps
Antenna coordinates	(0,0,1.5)
Communication range	250 m
Carrier sense range	550 m
Frequency	914 MHz
Receiving threshold	3.652e-10
Carrier sense threshold	1.559e-11
Capture threshold	10.0

Table 7-7: Simulation parameters of the 802.11 MAC/PHY.

Slot time	20 µs
SIFS	10 µs
DIFS	SIFS+2*Slots
PLCP preamble	144 bits
PLCP header	48 bits
Data rate	11 Mbps
PLCP data rate	1 Mbps ***
Basic data rate	1 Mbps
Antenna coordinates	(0,0,1.5)
Communication range	160 m
Carrier sense range	422.8 m
Frequency	2.4 GHz
Receiving threshold	1.15126e-10
Carrier sense threshold	1.559e-11
Capture threshold	10.0

*** 2Mbps in the standard

Table 7-8: Simulation parameters of the 802.11b MAC/PHY.

In the subsequent sections of the document we will refer to 802.11 – 1997 as simply 802.11 or 802.11 mode 0 and the modified 802.11b as 802.11b or 802.11 mode 1.

NS2 uses thresholds to determine whether one frame is received correctly. The carrier sense threshold (CSThresh_) determines whether frames are detected by the receiver or not. If the signal strength is lower than the specified threshold the frame is simply discarded in the physical layer. The receiving threshold determines if the frame is received correctly or it was corrupted then the Mac layer will discard it. The last threshold is the Capture threshold which determines if one of simultaneously received frames can be recuperated. It is calculated by the ratio of the strongest frame's signal strength to the signal strength sum of other frames.

Ns2 implements three propagation models, the free space propagation model, the two–ray ground reflection model and the shadowing model. The free space propagation model assumes the ideal propagation condition that there is only one clear line–of–sight path between the transmitter and receiver. The two–ray ground reflection model considers both the direct path and a ground reflection path. Finally the shadowing model considers more

complex effects of obstructions between the transmitter and receiver. It is mainly used to simulate wireless channels in in–door environments.

In our study we are interested to open environments of mobile nodes and the propagation model is the two–ray ground reflection model.

7.11.1.2. Number of hops effect on VoIP sessions support

Our first simulation tries to check the effect of the path length on the network delay and loss ratio within VoIP traffic. For this goal we worked on a string topology where we started with 2 nodes and until the loss ratio or network delay exceeds the required thresholds by VoIP flows. This experience is repeated with 5 different Codecs. The list of the selected set of Codecs and their attributes are shown in the table below.

Codec	GSM 6.10	G.711	G.723.1	G.726	G.729
Bit rate (Kbps)	13.2	64	6.3	32	8
Framing (ms)	20	20	30	20	10
Payload	33	160	24	80	10*2
Rate (packet/s)	50	50	33	50	50

Table 7-9: Selected Codecs and their attributes.

Our topology consists of a set of nodes placed on the same line and separated with a distance of 250 meters (the default communication range). We started with 2 nodes and increased the number of nodes to reach the level at the network becomes incapable to hold the VoIP traffic.

The obtained delay results with G.711 are shown on the graph (fig. 7.9). Over 2 hops the VoIP session becomes unfeasible and the network is incapable to support it.

The average network delay with G.711 is about 4.60 ms in a 1–hop network, 10.30 ms with 2 hops and 380 ms with 3 hops. The loss ration is 0 % with 1 and two hops but suddenly jump like the delay for 3 hops.

With the GSM 6.10 which generates a 13.2 kbps rate at the application level, the network offers better support until three hops. For 4 hops the VoIP packets receive extra delays and losses (fig. 7.10).

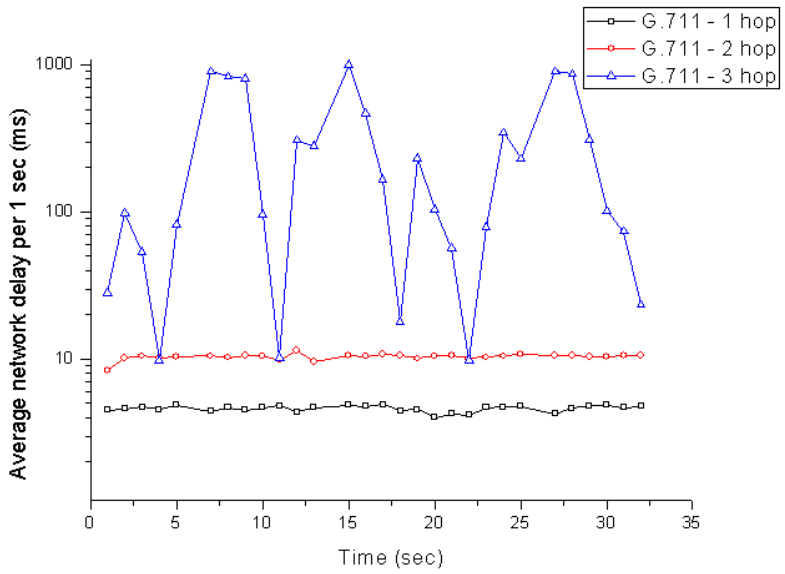


Figure 7-9: Network delay of VoIP traffic in a linear topology using G.711 – 802.11 (1997).

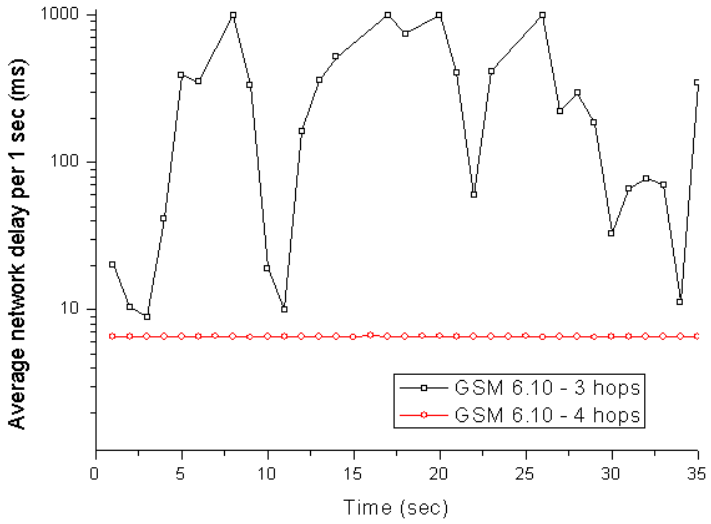


Figure 7-10: Network delay of VoIP traffic in a linear topology using GSM 6.10 – 802.11 (1997).

Finally, with the G.723.1 which generates a 6.3 kbps rate at the application level, the VoIP flow is well handled until 13 hops where the delay and loss become suddenly very high (fig. 7.11).

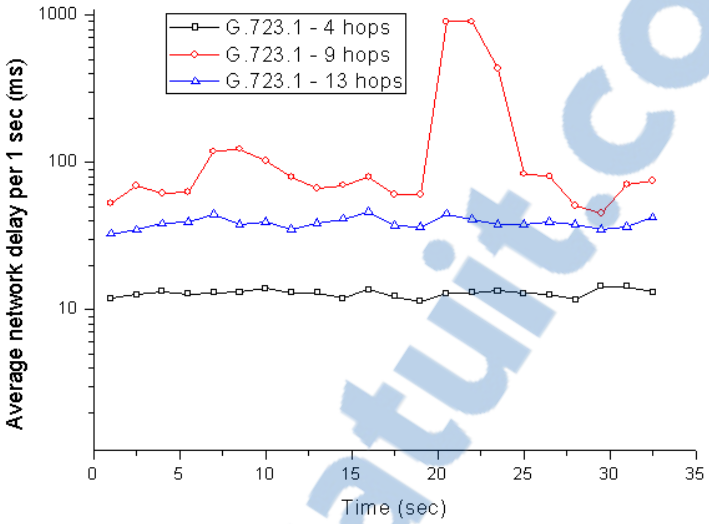


Figure 7-11: Network delay of VoIP traffic in a linear topology using G.723.1 – 802.11 (1997).

The average network delay with G.723.1 is about 13 ms in a 4–hops network, 38 ms with 9 hops and 300 ms with 13 hops where it explodes from time to time like shown in the graph around 15 seconds.

This shows that a VoIP session with a very low rate can be easily handled by ad hoc networks based on 802.11 and of a small size. Up to an acceptable number of hops (12 hops) the network delay and loss ratio stays low and respond the requirement of a VoIP application. However, for a voice codec, the higher is the compression the lower is the tolerance of errors by the coded frames and the bigger is the effect of the bit error rate (BER) on the loss ratio. This means that in a real environment the losses will to considerably increase and can make the use of such Codecs completely impractical.

The next graph (fig. 7.12) shows the evolution of the network delay in function of number of hops of the previously seen Codecs.

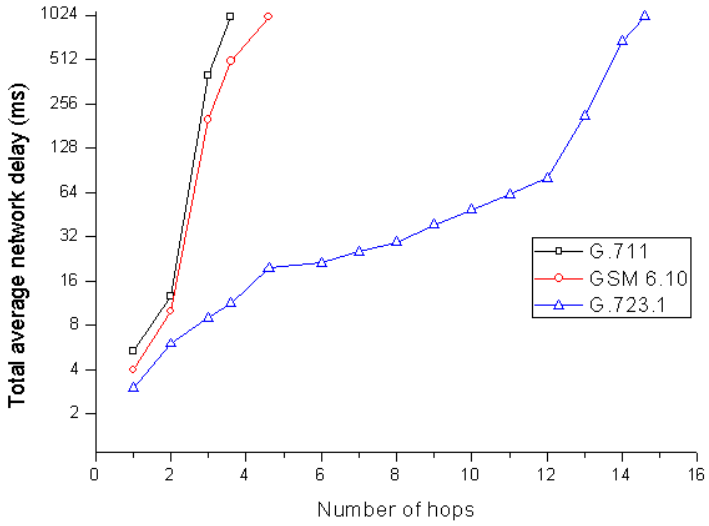


Figure 7-12: E2E delay of one VoIP session with different Codecs – 802.11.

The previous results give a clear idea on the difficulty to handle VoIP sessions in MANETs with the 802.11–1997 MAC/PHY layers. According to the same results we can only hope to have acceptable QoS in very small MANETs which is not of big interests for today applications.

After these first disappointing results one may ask if it really matters to think about multimedia applications on MANETs.

We then started new simulations with the 802.11b. It is an enhancement of the original 802.11. The initial 802.11 is no more manufactured while its replacement (802.11b) meets a big success especially for WLAN. The DSSS has been improved to include 5.5Mbps and 11Mbps data rates in addition to the 1Mbps and 2Mbps data rates of the initial standard. To provide the higher data rates, 802.11b uses CCK (Complementary Code Keying), a modulation technique that makes efficient use of the radio spectrum.

The receive threshold and the transmit power are set to match the ORiNOCO11b card specifications [114] (table 7.10).

Figure 7-13: the ORiNOCO 802.11b PC Card.

We are interested in open environments and the parameters shown in the table 7.8 are applied to get a range of 160 m. we repeated the same experiments we did with the initial 802.11 implementation.

Frequency BAND	**2.4 GHz**
Modulation	• CCK11 >>> 11Mbps • CCK5.5 >>> 5.5Mbps • QPSK >>> 2Mbps • BPSK >>> 1Mbps
BER	Better than 10^{-5}
Range	
Open Environment	160 m
Semi-Open Environment	50 m
Closed environment	25 m

Table 7-10: ORiNOCO11b PC card spec [114].

We observed that with the 802.11 the VoIP session with the G.711 drops if the path exceeds 2 hops. The same experience on the 802.11b shows a big difference. Indeed the delay is very low to a path length of 9 hops and the loss ratio is 0 %. At ten hops the traffic starts to receive bursts of high delays and losses (fig. 7.14).

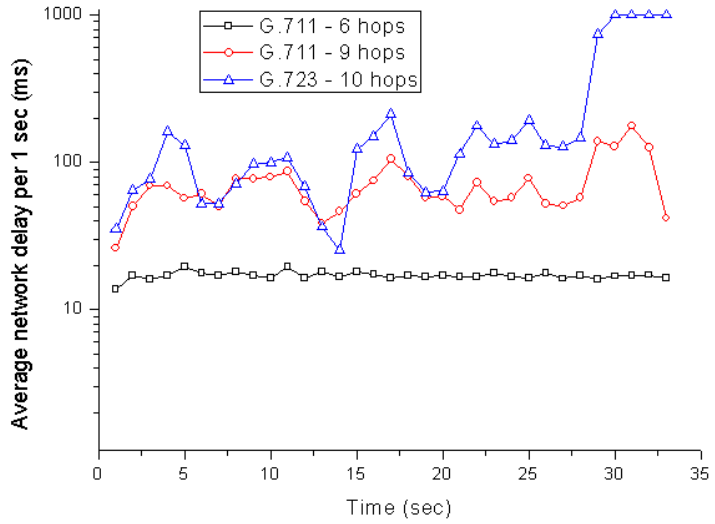


Figure 7-14: Network delay of VoIP traffic in a linear topology using G.711 – 802.11b.

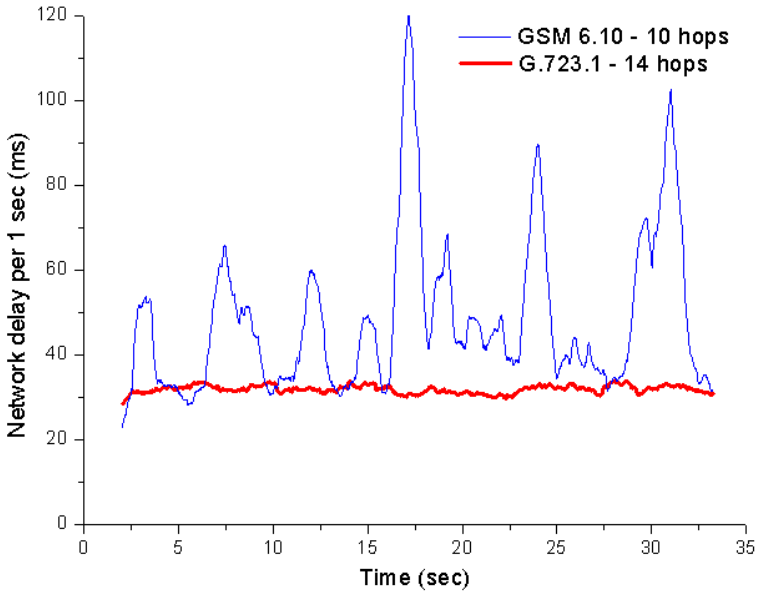


Figure 7-15: Network delay of a VoIP session with GSM 6.10 and G.723.1 – 802.11b.

With the GSM 6.10 which generates a 13.2 kbps rate at the application level and the G.723.1 which generate 6.3 kbps, the network offers good support. With a number of 10 hops the average network delay of a GSM 6.10 based session is about 48 ms with an upper limit of

120ms. With the G.723.1 the session is easily supported. Average delay of 30 ms is recorded for a path of 14 hops (fig. 7.15).

This shows that the 802.11b offer acceptable support of VoIP sessions for small and medium size ad hoc networks in ideal conditions. The evolution of the network delay in function of the number of hops of G.711, GSM 6.10 and G.723.1 is pointed up in the graph below.

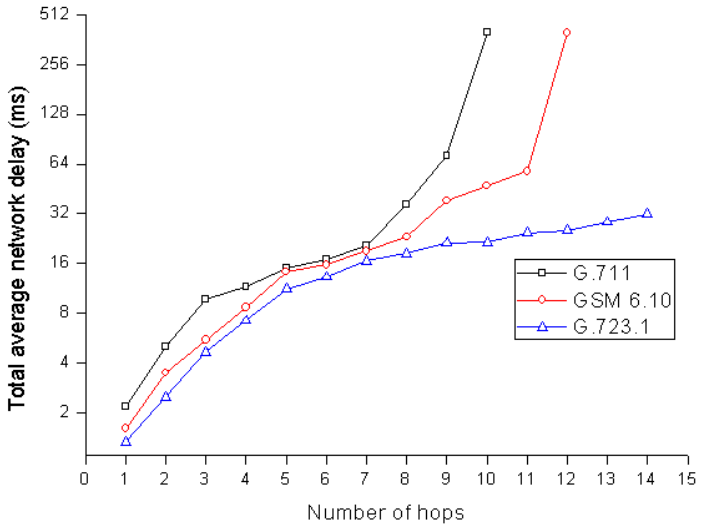


Figure 7-16: Network delay of one VoIP session with different Codecs – 802.11b.

The previous simulations results show that VoIP applications are initially supportable on small to medium size ad hoc networks with ideal conditions. Now we should try more realistic scenarios to verify how much those networks can endure real applications and environments conditions.

7.11.1.3. analyzing losses

Random packet loss is tolerated by VoIP applications. It is bursts of loss that degrade the perceived quality. A burst is a loss of consecutive packets. Human listeners don't complain if losses are randomly distributed. Bursts appear as momentary gaps in the conversation. If the gap is tiny then it can be accepted but large gaps make the conversation quality unacceptable. Let's take a look now on the losses recorded in the previous simulations. We remarked that the losses occur in bursts which interrupt the VoIP session

for long time. The graph in figure 7.17 show the network delay attained by VoIP packets with the 802.11 and on a 3 hops path. The bursts of losses are underlined with bold dashes in the graph.

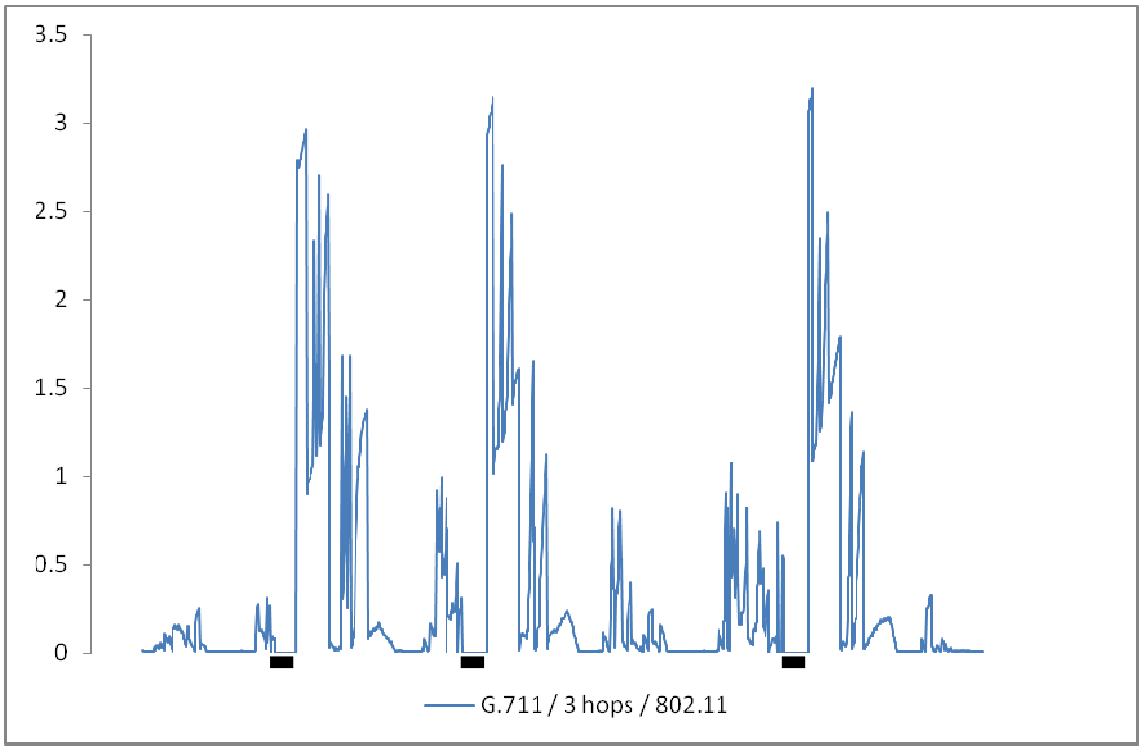


Figure 7-17: bursts of losses and network delay of a VoIP session.

Most bursts have a length of up to 700ms. This phenomenon happened with all the Codecs. When the number of hops reaches a threshold the delays suddenly jump to high values and the flows breaks for long time. Analyzing the simulator trace shows the misbehavior of the MAC layer which reports to the routing protocols that there is no route to the next hop in congested periods after few collisions. This forces the routing agent to repair the route and the same old route is often reused after the interruption. This problem can be lightened by the use of multipath routing or caching multiple routes to avoid routes recompilation delays which itself fuel the congestion in the network.

7.11.2. A more realistic scenario

Now after investigating the capacity of MANETs with the new 802.11b standard, we achieve additional simulations on a more realistic scenario with that standard (table 7.8).

7.11.2.1. Simulation settings

a. Topology and movement model

The simulated scenario considers a small ad hoc network of 30 nodes randomly distributed within an area of 650 x 250 like shown in figure 7.18.

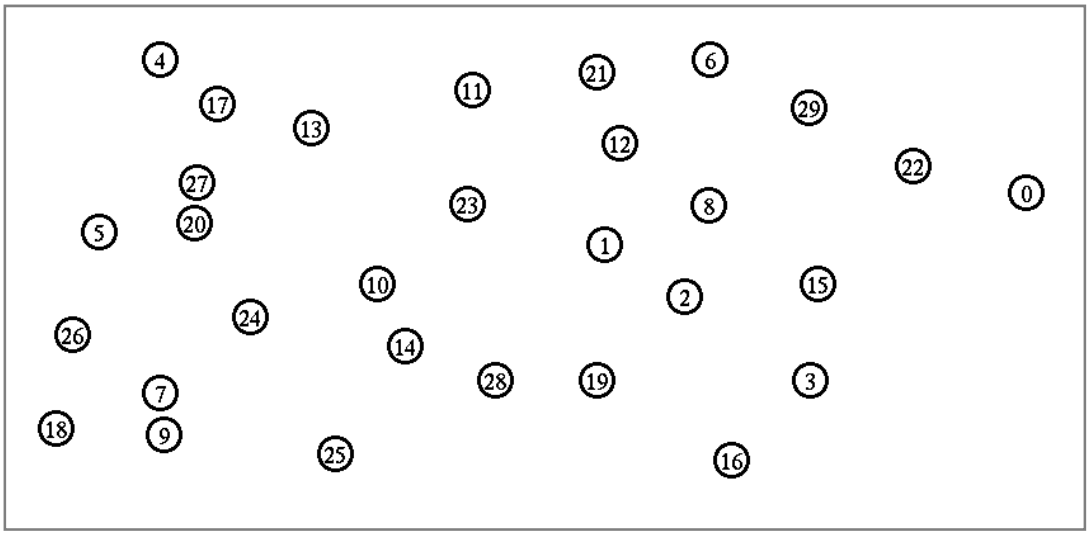
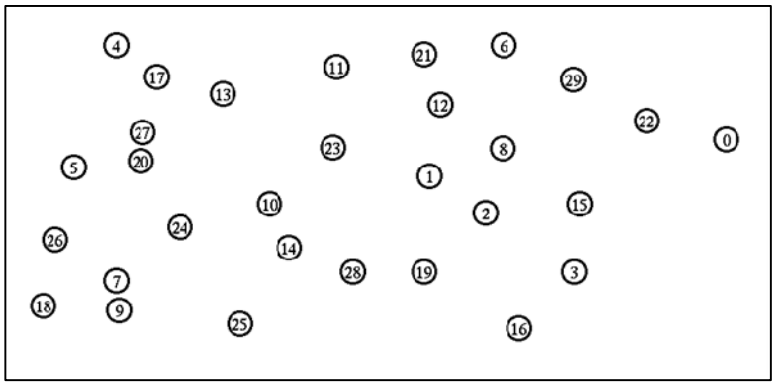


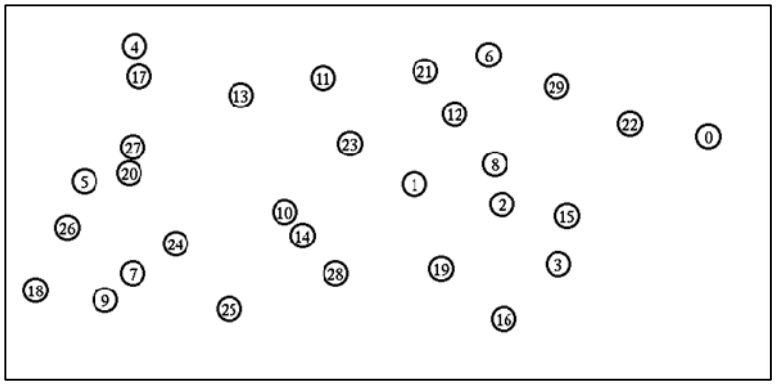
Figure 7-18: thirty nodes placed in a 650 x 250 m area.

Many mobility models have been proposed in the literature. The random walk model, the random waypoint model, the random direction model .... We used the random way point model to define nodes mobility. The Random Waypoint Mobility Model [115] includes pause times between changes in direction and/or speed. A node is either in pause or moving toward a destination with a random speed. Speed is uniformly distributed between provided MINSPEED and MAXSPEED. The Random Waypoint Mobility Model is the most used mobility model for ad hoc networks evaluation and has become a standard in mobile networking research. It can be described with 4 steps: pause, randomly select waypoint then move toward it with a random speed and repeat from the first step again. In our scenario we defined a pause time of 30 seconds and a speed interval of 1 – 5 m/s. The movement of nodes can be seen on the taken snapshot from nam (Network animator) in figure 7.19.

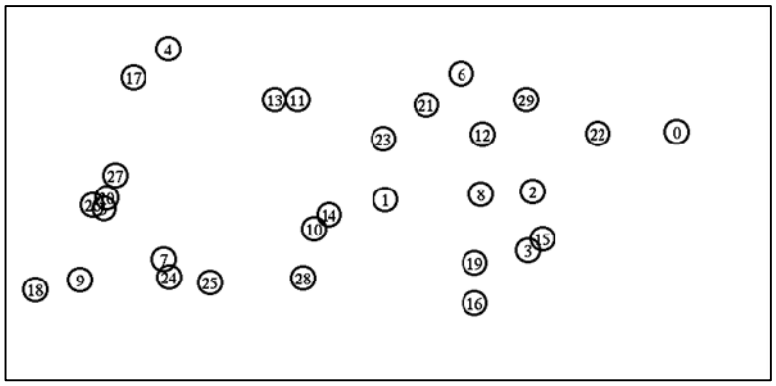
t = 0



t = 15



t = 40



t = 62

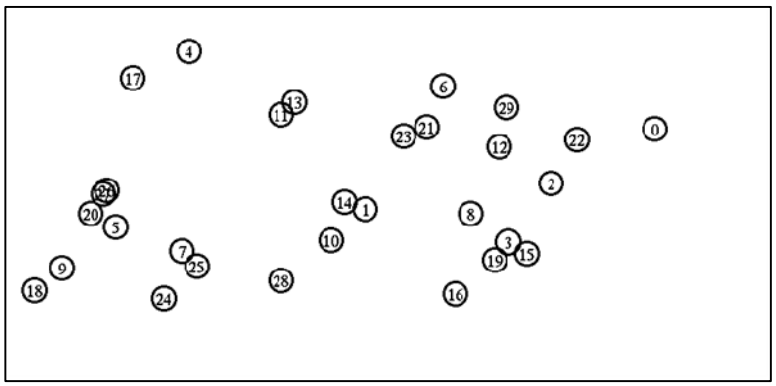


Figure 7-19: Nodes movement.

b. Traffic model

We added 2 VoIP sessions with the G.711 codec which generates a rate of 64k at the application level. The first session starts at time 2.0 seconds between nodes 4 and 15 which require a minimum of 4 hops and the second between nodes 11 and 22 starts at 4.0 seconds and need a minimum of three hops. These two sessions are not independent because of multiple intersections of nodes communication and carrier sense ranges along the paths.

7.11.2.2. Simulation results

Simulation parameters are set to satisfy the IEEE 802.11b specification of the standard at both physical and link layers as described in table 7.8. We measured delay and loss for each of the four VoIP flows. The figure 7.20 shows the obtained results for the VoIP flow from node 4 to node 15.

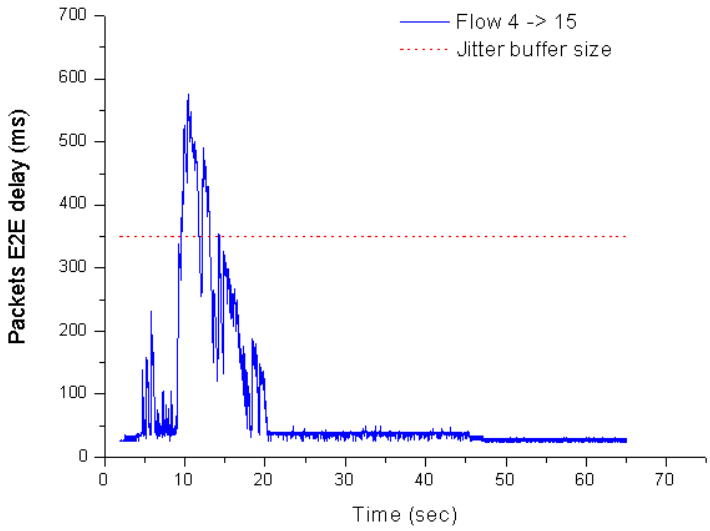


Figure 7-20: E2E delay vs. time of VoIP flow from node 4 to 15.

In this flow no loss has been recorded but many packets reach the destination and get dropped there because they arrive very late. For instance we assumed a fixed jitter buffer length of 350 ms and packets which exceed this delay are not useful. This threshold is marked by the dashed line on the graphs. This delay create a gap in the conversation of a length of about 2 seconds + 1 second just after and this is very unacceptable in the rules of voice conversations quality. After 45 seconds the delay becomes very low because the distance

between nodes become smaller with the movement and the needed hops decrease by one hop to become only three hops.

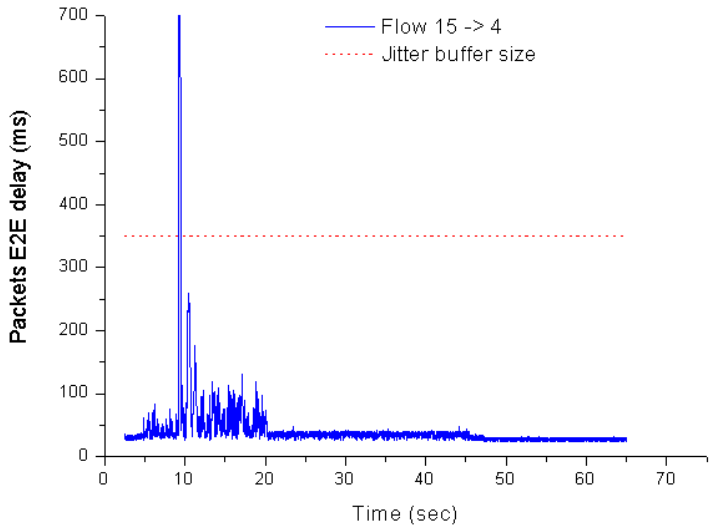


Figure 7-21: E2E delay vs. time of VoIP flow from node 15 to 4.

The opposite flow from node 15 to node 4 has received better handling. A smaller gap of 200 ms is recorded at the same moments. The analysis of the simulator trace shows that congestion at node 10 is behind this delay explosion. After 45 seconds the same remark as before apply here, the number of hops separating the two nodes becomes 3 with the nodes movement.

As distinguished in the first scenario the delay increase suddenly and this makes the prevention of such events harder or impossible if the carried traffic exceeds the capacity of the network.

The second VoIP session is also bidirectional and starts at time 4.0 seconds. The figure 7.22 shows the delay received by the packets of the flow from node 11 to 22.

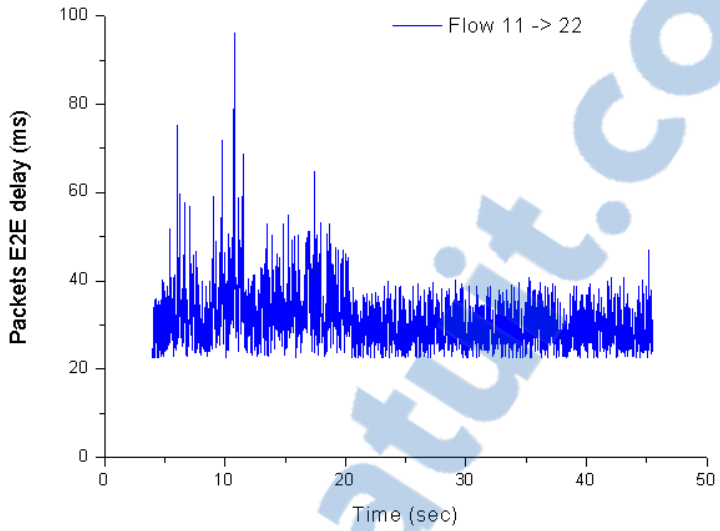


Figure 7-22: E2E delay vs. time of VoIP flow from node 11 to 22.

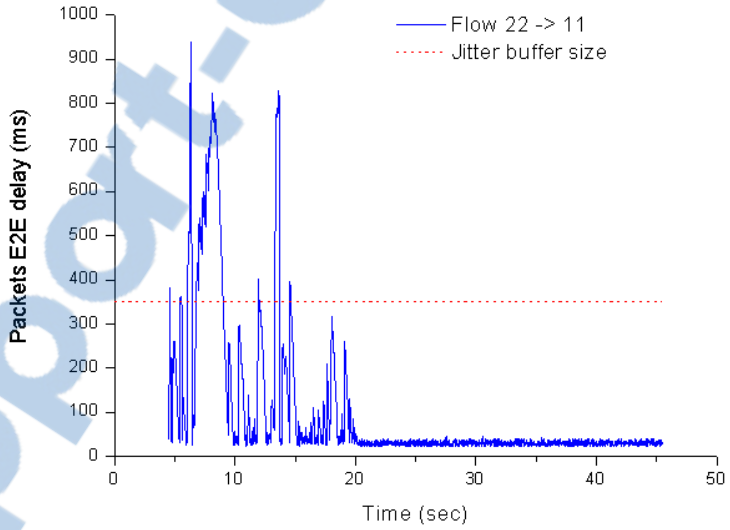


Figure 7-23: E2E delay vs. time of VoIP flow from node 22 to 11.

The first flow is well handled by the network and the maximal E2E delay is below 100 ms which means a very good voice quality. On the other hand in the opposite flow from node 22 to 11 we can distinguish three important delay explosion periods which results in three gaps of 300ms, 2.5s and 340ms.

Now we apply the EFORTS early packets dropping feature through the packets' deadline aware MAC on the same scenario to evaluate the advantage of our approach. The graphs below shows the old and the new E2E delay evolutions in function of simulation time.

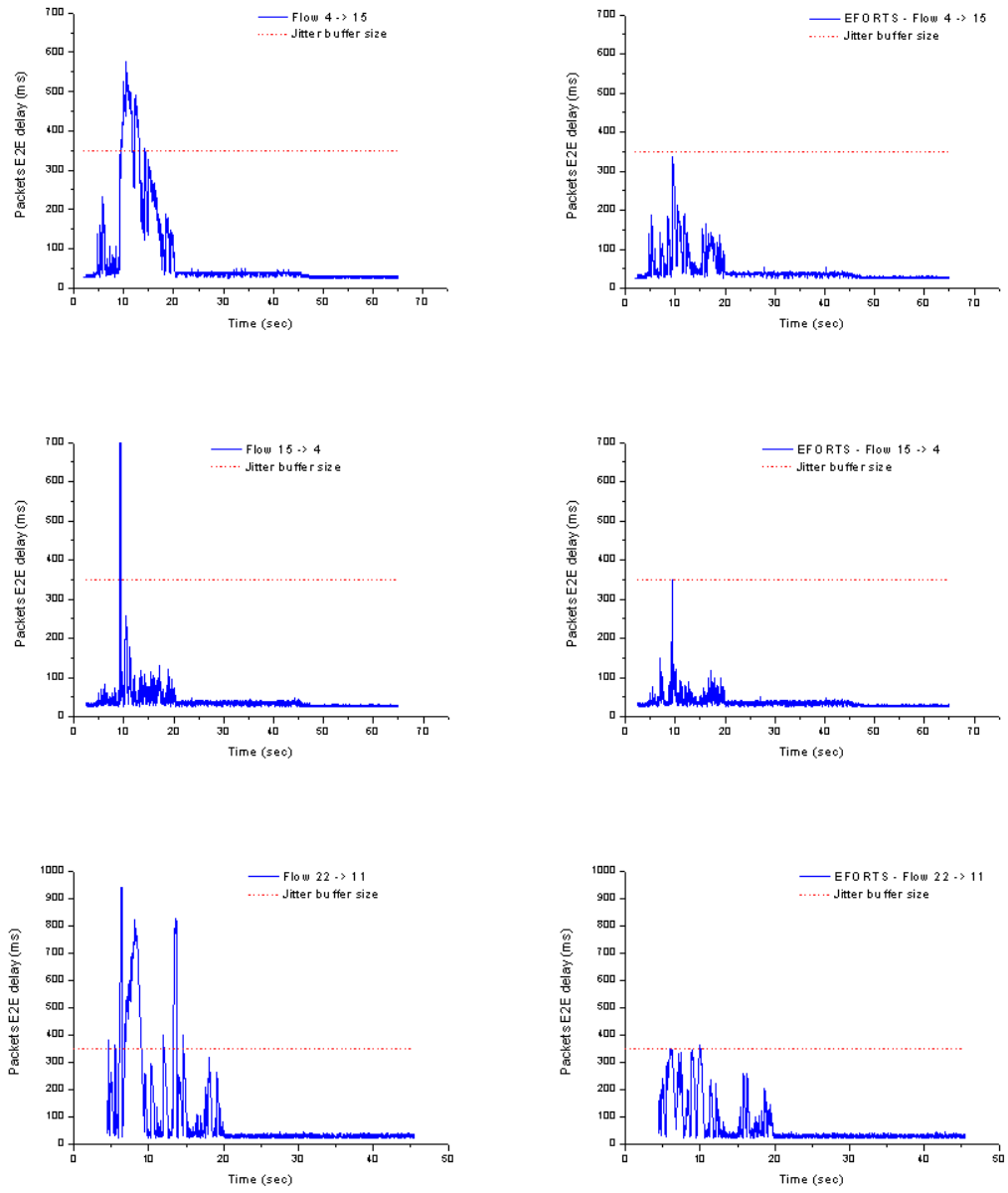


Figure 7-24: EFORTS deadline aware MAC performance, 802.11b vs EFORTS.

In the flow 4 to 15, with a fixed deadline of 350ms, the comparison between the two graphs reveals the reduction of the first gap length and the deletion of the second one. The table

below shows the quantitative difference between the two simulations: (1: 802.11b, 2: EFORTS deadline-aware MAC).

	4 to 15		15 to 4		11 to 22		22 to 11	
	1	**2**	**1**	**2**	**1**	**2**	**1**	**2**
Total average delay (ms)	77	**46**	43	**37**	31	**30**	101	**55**
Useful packets avg. delay (ms)	57	**46**	38	**37**	31	**30**	56	**55**
Loss ratio (%)	0	**0.06**	0	**0.32**	0	**0**	0	**4.5**
Exp. packets ratio (%)	5	**0**	0.5	**0**	0	**0**	8	**0.1**
Total loss (%)	5	**0.06**	0.5	**0.32**	0	**0**	8	**4.6**

Table 7-11: 802.11b vs. EFORTS deadline–aware MAC.

The table 7.11 shows that the dropped packets ratio has increased from 0 % to 0.06 % for the first flow. The first packets that exceed the maximal allowed delay are dropped. Thereby delays are bounded and useless packets are discarded before they reach the destination. This behavior has helped to reduce the length of the congestion period and consequently save additional packets from accumulating additional delays. This phenomenon is explained by the decreasing of the expiring packets that reached the node 15 from 5 % to only 0 % if we consider the dropped packets.

7.11.2.3. Further optimizations

In our simulations we considered a fixed maximal allowed delay. However it is possible to use dynamic deadlines in function of the jitter buffer size when using adaptive jitter buffers. We think this is very imperative after looking to the different results we got. To be more precise, if the delay of packet exceeds the current average by a determined amount then it will certainly exceed the fixed deadline. In this case we lost the opportunity to drop it earlier without saving it.

7.12.Conclusion

This chapter presents a cross layer optimization approach for performance enhancement of VoIP over ad hoc wireless networks. Performance improvement comes mainly from a healthier use of the network resources. Our first simulations focused on the benefits of the EFORTS deadline aware MAC and shows a promising improvement. However, we claim that ad hoc networks with today's technology are not ready to offer reliable multimedia applications. For this reason VoIP on ad hoc networks still need additional progresses to become really applicable. Big hopes are awaited especially from further technological progresses to offer enhancements of the physical layer capabilities. But also on all the layers a lot of improvements should and can be achieved in order to face the problems encountered.

If we knew what it was we were doing, it would not be called research, would it?
Albert Einstein

all I know is that I know nothing.
Socrates

Chapter 8

8. Conclusions and Future Work

8.1. Conclusions

The transmission of VoIP traffic over multihop ad hoc wireless networks is a very challenging research topic. If many works have addressed the performance problems of VoIP in WLANs, there are only few works which dealt with VoIP support in multihop ad hoc networks. This thesis considers the problem of VoIP quality in ad hoc networks and the effectiveness of cross layer designs in surmounting the limitations of conventional protocols.

We believe that cross layer design methodology allows better use of ad hoc networks' resources and hence can offer important optimizations of the VoIP applications support. We believe the solutions presented in this research thesis, namely EFORTS is an important contribution in the field of cross layer design and its usability to support multimedia applications in ad hoc networks. However, we should recall here that ad hoc network at the present time need more than architectural optimizations to reach an applicable level. We

want by that new technological advancement in the wireless communications field and voice coding. Special techniques are needed in that special environment. For instance techniques like silence suppression reduce considerably the amount of transmitted packets and should imperatively be supported by the Codecs for ad hoc networks.

Another other important conclusion we come to in our research work concern cross layer design. Even if cross layer design can lead to important optimizations, negative effects can come out and hence a cautionary approach should be followed. Also cross layer designs are more complex and require more detailed data on multiple levels. This will to delay the network development process and make the team work very fundamental to accelerate it. We experienced this in this work where we had to study many areas before we could envisage a contribution.

"The last conclusion I get is by concluding this work I realized that I have just started"

8.2. Future work

In our research work we gave big importance to the state of the art study. This allows us to propose EFORTS as a solution which optimizes handling of traffics with real time characteristics. As a step forward we envisage to add dynamic jitter buffer technique to our design and varying the packets deadline in function of this buffer evolution.

In addition to that we will try to achieve deeper simulations using different Codecs and with different routing protocols to get a more precise look on the problem facing VoIP quality in ad hoc networks.

We also envisage upgrading EFORTS toward a complete ad hoc networks optimization framework which have as a primary charge to optimize real time traffics support but also to solve the conflicts between the different applications classes and help the different layers in achieving wise resources utilization.

Finally, we pretend achieve experiments on real testbeds based on a network of PDAs or/and laptops. This step should allow us to get more consistent statistics.

REFERENCES

[01] I. Chlamtac, M. Conti, J. Liu, "Mobile Ad hoc Networking: Imperatives and Challenges", Ad Hoc Network Journal, Vol.1 N.1 January–February–March, 2003.

[02] S. Corson, J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", January 1999.

[03] M. Frodigh, P. Johansson, P. Larsson, "Wireless ad hoc networking: The art of networking without a network", ERICSSON review, issue n°04, 2000.

[04] Z. J. Haas, M. Gerla, D. B. Johnson, C. E. Perkins, "Guest Editorial, Wireless Ad Hoc Networks", IEEE Journal on Selected Areas in Communications, August, 1999.

[05] J. Liu, I. Chlamtac, "Mobile Ad Hoc Networking with a View of 4G Wireless: Imperatives and Challenges", IEEE press, 2004.

[06] P. Mohapatra, S. V. Krishnamurthy, "Ad Hoc Networks: Technologies and Protocols", Springer, 2005.

[07] P. Nicopolitidis, G. Papadimitriou, "Wireless networks", John Wiley & sons, ltd, 2003.

[08] R. Ramanathan, J. Redi, "A Brief Overview of Ad Hoc Networks: Challenges and Directions", IEEE Comm. Mag., vol. 40, no. 5, May. 2002.

[09] A. Goldsmith, "Wireless Communications", Cambridge University Press, 2005.

[10] C. Hedrick, "Routing Information Protocol", RFC1058, 1988.

[11] G. Malkin, "RIP version 2", RFC1723, Nov. 1998.

[12] S. Corson, J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC2501 January 1999.

[13] J. Moy, "OSPF version 2", RFC2328, April 1998.

[14] ISO, Intermediate System to Intermediate System intra–domain routing information exchange protocol for use in conjunction with the protocol for providing the connectionless–mode network service (ISO 8473), ISO/IEC 10589, 1992.

[15] C. E. Perkins, P. Bhagwat, "Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers", 1994.

[16] J. Broch, D. A. Maltz, D. B. Johnson, "A Performance Comparison of Multi–Hop Wireless Ad Hoc Network Routing Protocols", Computer Science Department, Carnegie Mellon University, 1998.

[17] V. D. Park, M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", Proceedings of IEEE INFOCOM '97, 1997.

[18] E. M. Gafni, D. Bertsekas, “Distributed algorithms for generating loop–free routes in networks with frequently changing topologies”, IEEE transactions on communications, January 1981

[19] D. B. Johnson, D. A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks”, 1996.

[20] D. Johnson, Y. Hu, D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4”, February 2007.

[21] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum, L. Viennot, “Optimized Link State Routing Protocol for Ad Hoc Networks,” 2001.

[22] T. Clausen, P. Jacquet, “Optimized link state routing protocol”, RFC 3626, 2003.

[23] C. E. Perkins, E. M. Royer, “Ad Hoc On–Demand Distance Vector Routing”, 1999.

[24] C. E. Perkins, E. Das, “Ad Hoc on–demand distance vector routing”. RFC 3561, July 2003.

[25] R. Ogier, F. Templin, "Topology Dissemination Based on Reverse–Path Forwarding (TBRPF)”, Feb. 2004.

[26] M. K. Marina, S. R. Das, “Routing In Mobile Ad Hoc Networks, Ad hoc networks: technologies and protocols”, Springer, 2005.

[27] M. Elizabeth “Routing Approaches In Mobile Ad Hoc Networks, Mobile Ad Hoc Networking”. IEEE Press and Wiley Interscience pub, 2004.

[28] R. Beraldi, R. Baldoni, “Unicast routing techniques for mobile Ad Hoc networks”, in “The handbook of mobile Ad Hoc networks” edited by Mohamed Ilyas, CRC Press, 2003.

[29] I. Chlamtac, M. Conti, J. Liu, “Mobile Ad Hoc Networking: Imperatives and Challenges”, Ad Hoc Network Journal, Vol.1 N.1, 2003.

[30] D. B. Johnson, “Routing in Ad Hoc Networks of mobile Hosts”, IEEE, 1995.

[31] J. Broch, D. A. Maltz, D. B. Johnson, “A Performance Comparison of Multi–Hop Wireless Ad Hoc Network Routing Protocols”, ACM/IEEE International Conference on Mobile Computing and Networking, 1998.

[32] C. Seguí, A. Zaballos, X. Cadenas, “Evolution of Unicast routing protocols in data networks”, 2004.

[33] L. M. Feeney, “A taxonomy for routing protocols in mobile ad hoc networks”, 1999.

[34] I. Chakeres, C. PERKINS, “Dynamic MANET On–demand (DYMO) Routing”, Internet–draft, july 2007.

[40] L. Kleinrock and F. Tobagi, “Packet switching in radio channels: Part ii – the hidden terminal problem in carrier sense multiple–access and the busy–tone solution,” IEEE Transactions on Communnication, December 1975.

[41] Raja Jurdak, Cristina Videira Lopes, and Pierre Baldi, “A survey, classification and comparative analysis of medium access control protocols for ad hoc networks,” IEEE Communications Surveys & Tutorials, First Quarter 2004.

[42] Sunil Kumar, Vineet S. Raghavan, Jing Deng, "Medium Access Control protocols for ad hoc wireless networks: A survey," November 2004.

[43] V. Kawadia, "protocols and architecture for wireless ad hoc networks", phd thesis, University of illinois at urbana–champaign, 2004.

[44] Bernhard H. Walke, Stefan Mangold, Lars Berlemann, "IEEE 802.11 wireless systems, protocols, Multihop/Mesh relaying, Performance and spectrum coexistence," John Wiley pub. 2006.

[45] N. Abramson, "The Aloha system – another alternative for computer communications," AFIPS Conf, 1970.

[46] Roberts L. G., "Aloha Packet System With and Without Slots and Capture," 1972.

[47] ETSI, "High performance radio local networks (HiperLAN) type 1 functional specification, HIPERLAN Functional Specification," ETSI standard, 1998.

[48] G. Anastasi, L. Lenzini, and E. Mingozzi, "Stability and performance analysis of HIPERLAN," Proc. IEEE INFOCOM 98.

[49] P. Karn, "MACA—a new channel access method for packet radio," 1990.

[50] V. Bhargavan, A. Demers, S. Shenker, L. Zhang, "MACAW—A Media Access protocol for wireless Lans, in: Proceedings of the ACM SIGCOMM," 1994.

[51] F. Talucci, M. Gerla, "MACA–BI (MACA By Invitation), A wireless MAC protocol for high speed ad hoc networking," 1997.

[52] Z.J. Haas, J. Deng, "Dual Busy Tone Multiple Access (DBTMA)—a multiple access control scheme for ad hoc networks," IEEE Trans. Commun., 2002.

[53] F. A.Tobagi and L. Klienrock, "Packet Switching in Radio Channels: Part II – The Hidden Terminal Problem in Carrier Sense Multiple Access and the Busy Tone Solution," IEEE Trans. Commun., 1975.

[54] C.–Shong Wu and V. O. K. Li, "Receiver–Initiated Busy Tone Multiple Access in Packet Radio Networks," Proc. ACM SIGCOMM'88.

[55] C. L. Fullmer and J. J. Garcia–Luna–Aceves, "Solutions to Hidden Terminal Problems in Wireless Networks," Proc. ACM SIGCOMM '97, vol.27, no.4, 1997.

[56] C. L. Fullmer and J. J. Garcia–Luna–Aceves, "Floor acquisition multiple access for packet–radio networks," Proc. ACM SIGCOMM '95.

[57] J. J. Garcia–Luna–Aceves, Chane L. Fullmer, "Floor acquisition multiple access (FAMA) in single–channel wireless networks", Mobile Networks and Applications, October 1999.

[58] Ajay Chandra, V. Gummalla, and J. O. Limb, "Wireless Medium Access Control Protocols," IEEE Comm., 2000.

[59] IEEE 802.11 WG, "Wireless Lan Medium Access Control (MAC) and Physical–Layer (PHY) specifications," 1999; standard.

[60] Hongqiang Zhai, Jianfeng Wang, Xiang chen and Yuguang Fang, "Medium access control in ad hoc networks: challenges and solutions," 2004.

[61] S. Shenker, R. Braden, and D. Clark. "Integrated services in the Internet architecture: an overview". Internet RFC 1633, June 1994.

[62] E. Crawley, R. Nair, B. Rajagopalan, H. Sandick, "A Framework for QoS–based Routing in the Internet", IETF RFC2386.

[63] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss 'RFC : 2475 "An Architecture for Differentiated Services", Request for Comments, IETF, December 1998.

[64] R. Braden, et al., "Resource ReSerVation Protocol (RSVP)", IETF RFC2205, Sep. 1997.

[65] H. Xiao, K.C. Chua and K.G. Seah, "Quality of Service Models for Ad–Hoc Wireless Network", ECE–ICR, Wireless Communications Laboratory, 2002.

[66] G. Ahn, A. T. Campbell, A. Veres, L. Sun, "SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks", INFOCOM, NY 2002.

[67] D. Chui and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks", computer networks, 1989.

[68] S.B. Lee, and A. T. Campbell, "INSIGNIA : In–band signaling support for QoS in mobile ad hoc networks", October 1998.

[69] R.O. Baldwin, N.J. Davis IV, S.F. Midkiff, "A real–time Medium Access Control protocol for ad hoc wireless local area networks", 1999.

[70] D.J. Deng, R.S. Chang, "A priority scheme for IEEE 802.11 DCF access method", 1999.

[71] Joao L. Sobrinho, A.S. Hrishnakumar, "Quality–of–Service in Ad Hoc Carrier Sense Multiple Access Wireless Networks", IEEE JSAC Aug 1999.

[72] IEEE 802.11 WG. "IEEE Std. 802.11e–2005, Amendment 8: Medium Access Control (MAC) Quality of Service (QoS) Enhancement", Sep. 2005.

[73] C. R. Lin, M. Gerla, "Asynchronous Multimedia Multihop Wireless Networks", IEEE INFOCOM 1997.

[74] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: a Core Extraction Distributed Ad Hoc Routing algorithm", IEEE Journal on Selected Areas in Communication, August 1999.

[75] G. Pei, M. Gerla, T.–W. Chen, "Fisheye State Routing in Mobile Ad Hoc Networks", Apr. 2000.

[76] C.E. Perkins, E.M. Royer, S.R. Das, "Quality of service for ad hoc on–demand distance vector routing", IETF Internet Draft, draft–ietf–manet–aodvqos–00.txt, nov 2001.

[77] S. Chen, K. Nahrstedt, "Distributed Quality–of–Service Routing in Ad–Hoc Networks", IEEE Journal on Special Areas in Communications, Vol. 17, No. 8, August 1999.

[78] T. Clausen, C. Dearlove, P. Jacquet, "The Optimized Link State Routing Protocol version 2", draft–ietf–manet–olsrv2–04, July 2007.

[79] S. Kopparty, S. V. Krishnamurthy, M. Faloutsos, S. K. Tripathi , “Split TCP for Mobile Ad Hoc Networks”, Department of Computer Science and Engineering, University of California, 2001.

[80] G. Dimic, N. D. Sidiropoulos, and R. Zhang, “Medium Access Control – Physical Cross–Layer Design,” IEEE Signal Processing Magazine, vol. 21, no. 5, 2004.

[81] L. Tong, V. Naware, and P. Venkitasubramaniam, “Signal Processing in Random Access,” IEEE Signal Processing Magazine, vol. 21, no. 5, Sept. 2004.

[82] M. Conti, G. Maselli, G. Turi, S. Giordano, “Cross–Layering in Mobile Ad Hoc Network Design”, IEEE Computer, 2004.

[83] K. Chandran, S. Raghunathan, S. Venkatesan and R. Prakash, “A Feedback–Based Scheme for Improving TCP Performance in Ad Hoc Wireless Networks”, IEEE Personal Communications Magazine, Vol. 8, No. 1, February 2001

[84] J. Liu, S. Singh, “ATCP: TCP for Mobile Ad–Hoc Networks”, IEEE Journal on Selected Areas in Communication, Vol. 19, No. 7, July 2001.

[85] S. Floyd, “TCP and explicit congestion notification”, ACM Computer Communication Review, Oct. 1994.

[86] Dzmitry Kliazovich, “Cross–Layer performance optimization in wireless local area networks”, Phd thesis, DIT – University of Trento, December 2006.

[87] Fabrizio Granelli, Michael Devetsikiotis, “Designing cross–layering solutions for wireless networks: a general framework and its application to a voice–over–WiFi scenario”, Trento Univ., Italy, 2006.

[88] V. Srivastava, M. Motani, “Cross–Layer Design: a survey and the road ahead, Communication magazine”, IEEE, Vol 43, Issue 12, Dec 2005.

[89] R. Winter, J. Schiller, N. Nikaein, and C. Bonnet, “CrossTalk: A Data Dissemination–based Cross–layer Architecture for Mobile Ad–hoc Networks”, 2005.

[90] J. Postel, “Internet control message protocol”, RFC 792, September 1981.

[91] G. Holland, N. Vaidya, and P. Bahl, “A Rate–Adaptive MAC Protocol for Multi–hop Wireless Networks”, 2001.

[93] Fernando Dıaz de Marıa, “Voice over IP and Wireless: Principles and Challenges”, edited in “Signal processing for mobile communications handbook” by Mohamed Ibnkahla, CRC Press, 2005.

[94] ITU–T Recommendation G.114, “One Way Transmission Time”. ITU–T, May 2003.

[95] Elias Nemer, “Handling VoIP Speech Coding Challenges”, Intel, 2002.

[96] ITU–T Recommendation P.59 (1993), “Artificial Conversational Speech”, 1993.

[97] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, “Cross–Layer Design for Wireless Networks,” IEEE Communications Magazine, vol. 41, Oct. 2003.

[98] ITU–T, "Methods for Subjective Determination of Voice Quality (ITU–T P.800)", 2006.

[99] ITU G.109, "Definition of categories of speech transmission quality", 1999.

[100] ITU G.107, "The E–model, a computational model for use in transmission planning", 2005.

[101] ITU–T Recommendation G.711, "Pulse Code Modulation (PCM) of Voice Frequencies", November 1988.

[102] ITU–T Recommendation G.729 Annex A., "C Source Code and Test Vectors for Implementation Verification of the G.729 Reduced Complexity 8 kbit/s CS–ACELP Speech Codec", ITU–T, November 1996.

[103] ITU–T Recommendation G.723.1, "Speech Coders: Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s", 1996.

[104] ITU–T Recommendation H.323 version 4, "Packet–based multimedia communications systems," ITU, Nov. 2000.

[105] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "IETF RFC 3261," SIP: Session initiation protocol, June 2002.

[106] H. Schulzrinne, R. Frederick, V. Jacobson, RTP: A Transport Protocol for Real–Time Applications, Request for Comments: 3550, July 2003.

[107] ITU–T Recommendation P.862, "Perceptual evaluation of speech quality (PESQ), an objective method for end–to–end speech quality assessment of narrowband telephone networks and speech codecs", 2001.

[108] V. T. Raisinghani and S. Iyer, "Cross–Layer Design Optimizations in Wireless Protocol Stacks," 2004.

[109] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, "SIP: Session Initiation Protocol", Request for Comments: 3261, June 2002.

[110] Ahmed MEDDAHI, H. Afifi, "Packet–E–Model: e–model for VoIP quality evaluation", 2006.

[111] Henry Sinnreich, Alan B. Johnston, "Internet Communications Using SIP, Delivering VoIP and Multimedia Services with Session Initiation Protocol", Second Edition, p82, 2006.

[112] J. Rosenberg, J. Lennox H. Schulzrinne, "Programming Internet Telephony Services", 1999.

[113] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross–layer design", IEEE Wireless Communication Magazine, 2004.

[114] ORiNOCO 11b Client PC Card Specification, Lucent technologies – Bell labs, 2004.

[115] T. Camp, J. Boleng, V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research", (WCMC), 2002.

[116] Intel white paper, "Overcoming Barriers to High-Quality Voice over IP Deployments", 2003.

[117] "The network simulator – ns2", in http://www.isi.edu/nsnam/ns/.