

Álvaro Torres Cortés
atcortes@batousay.com

Ph.D Advisors:
Pietro Manzoni
Carlos T. Calafate

Efficient real-time video delivery in vehicular networks



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Grupo de Redes de Computadores.
Departamento de Informática de Sistemas y Computadores
Universitat Politècnica de València.

Valencia, España. Noviembre de 2015

Dedicado especialmente a mis padres y a Cris.
ほんとうにありがとうございました。

“To infinity and beyond!”

Acknowledgments

As “*There is no emoticon for what I’m feeling*” I’ll try to express it with words.

During these years I can’t only say that I’ve been working, but also that I’ve been living with the GRC family.

In the first place I have to express my most sincere gratitude to my two PhD. advisors, Professors Manzoni and Calafate.

Carlos will always have my most sincere gratitude and respect, in the first place for his kindness, but specially for his ability of defying the laws of physics as his corrections are faster than light.

Pietro “*El padrino*”, with infinite patience has been able to cope with my “*I have a little problem*” catch phrase, and has supported my work during all these years, not only with ideas but also making the impossible to obtain proper equipment.

Juan Carlos, always cheering, always pushing, maintaining a high group morale, has always been a great help.

I have to thank Johann, you taught me and helped me a lot in this research world, thanks for your tips, advise and friendship.

Sergio and Miguel, I've enjoyed every moment of working with you, our chats and coffee breaks, and as you know, *You're next!*

As for the rest of the past and present GRC family (the 5 Jorges, Ali, Shuba, Oscar, Marga, ...), thank you everyone for your kindness.

Also, I have to thank the people I met during one of the best periods of my PhD, my stay in Tokyo. In the first place to Professor Ji for accepting me in her research group. In the second place to Fernando, Alberto, Sánchez and Hugo, you made me feel like I was at home!

At last but not least, I have to thank my parents Ramón and Isabel, and my girlfriend Cristina, thanks for always being there and your unlimited support.

This is not only the end of a chapter, but the beginning of a new one.

Contents

List of Figures	13
Abstract	21
Resumen	23
Resum	25
1 Introduction	27
1.1 Technologies employed	28
1.1.1 Wireless communications overview	29
1.1.2 Digital video coding techniques overview	33
1.2 Challenges	43
1.3 Methodology	44

1.4	Objectives	45
1.5	Organization of the thesis	46
2	Publication A:	
	Deploying a real IEEE 802.11e testbed to validate simulation results	47
2.1	Introduction	48
2.2	Related Works	50
2.3	Overview of the IEEE 802.11 Standard	51
	2.3.1 IEEE 802.11e: MAC enhancements for QoS	53
2.4	Testbed Setup	56
	2.4.1 Hardware and software details	58
	2.4.2 Throughput and delay measurements	59
2.5	Experimental Results	60
	2.5.1 Traffic differentiation analysis	61
	2.5.2 QoS robustness analysis	65
2.6	Conclusions and Future Work	68
3	Publication B:	
	Assessing the IEEE 802.11e QoS effectiveness in multi-hop indoor scenarios	71
3.1	Introduction	72
3.2	Related Works	74
3.3	Overview of the IEEE 802.11 Standard	76
	3.3.1 IEEE 802.11e: MAC enhancements for QoS	77
3.4	Testbed Setup	80

3.4.1	Hardware and software details	80
3.4.2	Throughput and delay measurements	81
3.5	Comparative analysis of simulation and testbed results	83
3.5.1	Scenario definition	84
3.5.2	Traffic differentiation analysis	86
3.5.3	QoS robustness analysis	88
3.6	Evaluating IEEE 802.11e in a realistic indoor scenario	90
3.6.1	Node deployment and configuration issues	91
3.6.2	Performance results	93
3.7	Conclusions and Future Work	97
4	Publication C:	
	Distributed Admission Control in 802.11e-Based MANETs:	
	from Theory to Practice	101
4.1	Introduction	102
4.2	DACME Overview	104
4.3	Testbed Setup	106
4.4	Experimental results	107
4.5	Conclusions and future work	108
5	Publication D:	
	Evaluating the Effectiveness of a QoS Framework	
	for MANETs in a Real Testbed	111
5.1	Introduction	112
5.2	Related works	114
5.3	DACME Overview	115

5.3.1	Admission decision algorithms meeting bandwidth requirements	116
5.3.2	Admission decision algorithms meeting delay requirements	119
5.3.3	Implementation details	120
5.4	Testbed setup	122
5.5	Experimental Results	123
5.5.1	Varying the number of Best Effort flows	124
5.5.2	Varying the number of Video Flows	126
5.5.3	Varying the maximum delay restriction	128
5.6	Conclusions and future work	130

6	Publication E:	
	Evaluating H.265 Real-Time Video Flooding Quality in Highway V2V Environments	133
6.1	Introduction	134
6.2	Related Work	135
6.3	Overview of the video delivery strategy	137
6.3.1	Counter-Based strategy	138
6.3.2	Distance-Based strategy	138
6.4	Simulation Environment	139
6.5	Results	141
6.5.1	Tuning the flooding algorithms	142
6.5.2	Evaluating the best codec / flooding algorithm combination	145
6.5.3	Error resilience and impact on the video quality	146
6.6	Conclusions and future work	150

7	Publication F:	
	V2X Solutions for Real-time Video Collection	151
7.1	Introduction	152
7.2	Related work	154
7.3	Description of the flooding schemes	156
	7.3.1 Basic schemes	156
	7.3.2 Adapted schemes	158
7.4	Scenario and Methodology.	159
	7.4.1 Scenario	161
	7.4.2 Methodology	162
7.5	Results	163
7.6	Conclusions and future work	166
8	Publication G:	
	Evaluation of Flooding Schemes for Real-time Video Transmission in VANETs	169
8.1	Introduction	170
8.2	Related work	172
8.3	Description of the flooding schemes	176
	8.3.1 Basic schemes	177
	8.3.2 Adapted Schemes	178
	8.3.3 Proposed Schemes	179
8.4	Scenario and Methodology.	183
	8.4.1 Simulation Scenario	185
	8.4.2 Methodology	186

8.5	Results	192
8.5.1	Comparison of Flooding Schemes	192
8.5.2	Impact of GPS Drift	193
8.5.3	Urban scenario	195
8.6	Conclusions	197
9	Results' discussion	209
10	Conclusions	227
11	Summary of publications	229
	Funding Acknowledgments	233
12	Bibliography	235

List of Figures

1.1	Detail of CSMA/CA mechanism.	30
1.2	IFS, CWmin and CWmax for each traffic category. .	32
1.3	Evolution of the ITU-T and MPEG standards. . . .	34
1.4	Types of frames. (Source: Wikimedia Commons) .	36
1.5	H.264 Intra Prediction.	37
1.6	H.264 Macroblock	38
1.7	H.265 Intra Prediction	39
1.8	H.265 Coding Tree Block example.	40
2.1	Multi-hop ad-hoc network scenario used for testing.	56
2.2	Load vs. delay values for the Fast-Ethernet link used as return path.	60
2.3	Throughput values achieved via simulation (top) and in our testbed (bottom).	62

2.4	End-to-end delay values achieved via simulation (top) and in our testbed (bottom).	64
2.5	Throughput stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (top) and from the testbed (bottom).	66
2.6	End-to-end delay stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (top) and from the testbed (bottom).	67
3.1	Load vs. delay values for the Fast-Ethernet link used as return path.	83
3.2	Multi-hop ad-hoc network scenario used for testing.	84
3.3	Throughput values achieved via simulation (left) and in our testbed (right).	87
3.4	End-to-end delay values achieved via simulation (left) and in our testbed (right).	88
3.5	Throughput stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).	89
3.6	End-to-end delay stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).	90
3.7	Map of the building floor where tests were made and node deployment details. Approximate floor dimensions (maximum): 60×40 meters.	92

LIST OF FIGURES

3.8	UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at one hop using static (left) and dynamic (right) routing.	94
3.9	UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at two hops using static (left) and dynamic (right) routing.	95
3.10	UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at three hops using static (left) and dynamic (right) routing.	96
3.11	UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at four hops using static (left) and dynamic (right) routing.	97
4.1	DACME diagram illustrating the dependencies among the different elements involved.	105
4.2	Chain topology used for testing.	106
4.3	Mean bit rate when flows are active (top) and mean end-to-end delay (bottom).	109
5.1	Bandwidth estimation strategy.	116
5.2	Delay estimation method.	120
5.3	DACME diagram illustrating the dependencies among the different elements involved.	121
5.4	Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the number of background traffic sources. . .	124

5.5	Mean values for throughput (left) and end-to-end delay (right) when varying the number of background traffic sources.	126
5.6	Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the number of concurrent video sources. . .	127
5.7	Mean values for throughput (left) and end-to-end delay (right) when varying the number of concurrent video sources.	127
5.8	Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the maximum delay restriction.	129
5.9	Mean values for throughput (left) and end-to-end delay (right) when varying the maximum delay restriction.	130
6.1	Overview of the simulated scenario.	139
6.2	Counter-Based (left) vs. Distance-Based (right). Vehicle arrival rate of 1s.	143
6.3	Counter-Based (left) vs. Distance-Based (right). Vehicle arrival rate of 2s.	144
6.4	Counter-Based (left) vs. Distance-Based (right). Vehicle arrival rate of 3s.	145
6.5	Packet delivery ratio. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).	147
6.6	Frame loss. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).	148
6.7	PSNR. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).	149

LIST OF FIGURES

7.1	Highway scenario.	161
7.2	High density scenario. Percentage of packets received.	163
7.3	Medium density scenario. Percentage of packets received.	164
7.4	Low density scenario. Percentage of packets received.	165
8.1	Inheritance diagram of the different flooding schemes.	176
8.2	Highway scenario.	185
8.3	Urban scenario.	187
8.4	Overview of the video quality measurement.	188
8.5	GPS error emulation.	190
8.6	High density scenario. Percentage of packets received (up) and PSNR (down).	199
8.7	Medium density scenario. Percentage of packets received (up) and PSNR (down).	200
8.8	Low density scenario. Percentage of packets received (up) and PSNR (down).	201
8.9	High density scenario. Percentage of packets received (up) and PSNR (down).	202
8.10	Medium density scenario. Percentage of packets received (up) and PSNR (down).	203
8.11	Low density scenario. Percentage of packets received (up) and PSNR (down).	204
8.12	Medium density scenario. Percentage of packets received. Positioning error comparison.	205
8.13	Cumulative Distribution Function for packet loss in both scenarios.	205

8.14	Urban High density scenario ($200\text{vehicles}/\text{km}\hat{A}^2$). Percentage of packets received (up) and PSNR (down).	206
8.15	Urban Medium density scenario ($100\text{vehicles}/\text{km}\hat{A}^2$). Percentage of packets received (up) and PSNR (down).	207
8.16	Urban Low density scenario ($50\text{vehicles}/\text{km}\hat{A}^2$). Per- centage of packets received (up) and PSNR (down).	208
9.1	Throughput values achieved via simulation (left) and in our testbed (right).	212
9.2	End-to-end delay values achieved via simulation (left) and in our testbed (right).	213
9.3	Throughput stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).	214
9.4	End-to-end delay stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).	215
9.5	UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at three hops using static (left) and dynamic (right) routing.	216
9.6	Mean bit rate when flows are active (left) and mean end-to-end delay (right).	217
9.7	Mean values for throughput (left) and end-to-end delay (right) when varying the number of concurrent video sources.	219
9.8	Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the number of concurrent video sources.	219

LIST OF FIGURES

9.9	Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the maximum delay restriction.	220
9.10	Packet delivery ratio. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).	223
9.11	High density scenario. Percentage of packets received.	224
9.12	Low density scenario. Percentage of packets received (left) and delay (right).	224
9.13	High density scenario. Percentage of packets received (left) and PSNR (right).	225
9.14	Urban High density scenario ($200\text{vehicles}/\text{km}^2$). Percentage of packets received (left) and PSNR (right).	226

Abstract

Vehicular Ad-hoc Networks (VANET) are a special type of networks where the nodes involved in the communication are vehicles. VANETs are created when several vehicles connect among themselves without the use of any infrastructure. In certain situations the absence of infrastructure is an advantage, but it also creates several challenges that should be overcome.

One of the main problems related with the absence of infrastructure is the lack of a coordinator that can ensure a certain level of quality in order to enable the correct transmission of video and audio. Video transmission can be extremely useful in this type of networks as it can be used for videoconferencing of by traffic authorities to monitor the scene of an accident.

In this thesis we focused on real time video transmission, providing solutions for both unicast and multicast environments. Specifically, we built a real-world testbed scenario and made a comparison with simulation results to validate the behavior of the simulation models.

Using that testbed we implemented and improved DACME, an admission control module able to provide Quality of Service (QoS) to unicast video transmissions. DACME proved to be a valid solution to obtain a certain level of QoS in multi-hop environments.

Concerning multicast video transmission, we developed and simulated several flooding schemes, focusing specifically on VANET environments. In this scope, the main contribution of this thesis is the Automatic Copies Distance Based (ACDB) flooding scheme. Thanks to the use of the perceived vehicular density, ACDB is a zeroconf scheme able to achieve good video quality in both urban and highway environments, being specially effective in highway environments.

Resumen

Las redes vehiculares ad-hoc (VANET) son un tipo especial de redes en las que los nodos que participan de la comunicación son vehículos. Las VANETs se crean cuando diversos vehículos se conectan entre ellos sin el uso de ninguna infraestructura. En determinadas situaciones, la ausencia de infraestructura es una ventaja, pero también crea una gran cantidad de desafíos que se deben superar.

Uno de los principales problemas relacionados con la ausencia de infraestructura, es la ausencia de un coordinador que pueda asegurar un determinado nivel de calidad, para poder asegurar la correcta transmisión de audio y vídeo. La transmisión de vídeo puede ser de extrema utilidad en este tipo de redes ya que puede ser empleada para videoconferencias o por las autoridades de tráfico para monitorizar el estado de un accidente.

En esta tesis nos centramos en la transmisión de vídeo en tiempo real, proveyendo soluciones tanto para entornos unicast como multicast. En particular construimos un banco de pruebas real y

comparamos los resultados obtenidos con resultados obtenidos en un entorno simulado para comprobar la fiabilidad de estos modelos.

Usando el mismo banco de pruebas, implementamos y mejoramos DACME, un módulo de control de admisión capaz de proveer de calidad de servicio a transmisiones de vídeo unicast. DACME probó ser una solución válida para obtener ciertos niveles de calidad de servicio en entornos multisalto.

En lo referente a la transmisión de vídeo multicast, desarrollamos y simulamos diversos esquemas de difusión diseñados específicamente para entornos VANET. En este campo, la principal contribución de esta tesis es el esquema de difusión “*Automatic Copies Distance Based*” (ACDB). Gracias al uso de la densidad vehicular percibida, ACDB es un esquema, que sin necesidad de configuración, permite alcanzar una buena calidad de vídeo tanto en entornos urbanos como en autopistas, siendo especialmente efectivo en este último entorno.

Resum

Les xarxes vehiculars ad-hoc (VANET) son un tipus de xarxes especials a les que els diferents nodes que formen part d'una comunicació son vehicles. Les VANETs es formen quan diversos vehicles es connecten sense fer ús de cap infraestructura. A certes situacions l'absència d'una infraestructura suposa un avantatge, encara que també genera una gran quantitat de desafiaments que s'han de superar.

U dels principals problemes relacionats amb l'absència d'infraestructura, és la manca d'un coordinador que pugui garantir una correcta transmissió tant de video com d'àudio. La transmissió de video pot ser d'extrema utilitat a aquest tipus de xarxes, ja que es pot emprar tant per a videoconferències com per part de les autoritats de trànsit per monitoritzar l'estat d'un accident.

A aquesta tesi ens centrem en transmissió de video en temps real, proporcionant solucions tant a entorns unicast como a entorns multicast. Particularment, vam construir un banc de proves i obtinguérem resultats que comparàrem amb resultats obtinguts mit-

jançant simulació. D'aquesta manera validarem la fiabilitat dels resultats simulats.

Fent ús del mateix banc de proves, vàrem implementar i millorar DACME, un mòdul de control d'admissió, capaç de proveir de qualitat de servici a transmissions de video unicast. DACME va provar ser una bona solució per obtindre un bon nivell de qualitat de servici en entorns de xarxes ad-hoc amb diversos salts.

Si ens centrem a la transmissió de video multicast, vàrem implementar i simular diferents esquemes de difusió, específicament dissenyats per al seu ús a entorns VANET. La principal contribució d'aquesta tesi es l'esquema de difusió ACDB (Automatic Copies Distance Based). Fent ús de la densitat vehicular, ACDB es capaç d'obtindre una bona qualitat de video tant a ciutats com a vies interurbanes, sent a especialment efectiu a aquestes últimes. A més a més no es necessària cap configuració per part de l'usuari.

1

Introduction

Recent advances in vehicular communication and Intelligent Transportation Systems (ITS) have attracted the interest of both the research community and the industry.

In particular, applications that include communication between vehicles have the potential of reducing the number of accidents and, therefore, of saving lives. In case an accident happens, they can also reduce both the severity and the physical consequences of the accident, while simultaneously reducing the economic impact of the accidents.

Nowadays, these advances are being presented in the mass media, generating high expectations in the society. Big companies are showing their autonomous-driving vehicle prototypes. These vehicles require of a high number of sensors, cameras, extremely precise GPS, etc. One of the necessary additions to these vehicles is wireless connectivity with the purposes of traffic coordination, route selection and pollution minimization. A good example of an advanced prototype is convoy driving, where a truck can lead a group of several autonomous driving vehicles in highway routes.

For these applications, ad-hoc wireless connectivity is required.

A first step into these new driving paradigms are initiatives like the European eCall, intended to bring rapid assistance to people involved in a collision anywhere in the European Union, which show the importance for governments of reducing fatalities on the road, as well as the importance of providing a fast response to accidents.

The work presented in this thesis is also oriented at improving the response upon accidents and providing better support to rescue teams. However, instead of focusing on accident warning, we focus on real-time video communication between vehicles or rescue team members. Vehicular communication can take place between two defined vehicles (unicast), or from one vehicle to many (multicast). Multicast real-time video transmission is intended for safety reasons after a crash, so traffic authorities can instantly evaluate the road conditions and assign the necessary resources. Instead, unicast video transmission is intended for communication between rescue team members, that is, when firefighters, ambulances and the police need to coordinate themselves on their way to, or at the crash site.

Specifically, this thesis focuses on the feasibility of real-time video transmission over ad-hoc networks, including both pedestrian and vehicular environments, evaluating the expected quality and several techniques to make it possible.

1.1 Technologies employed

In this subsection we will briefly describe the technologies employed in the development of the different works associated to this thesis.

In the first place, we will explain the IEEE 802.11 wireless communications standard with an emphasis on its quality of service

extension (IEEE 802.11e), and also the extension specific to vehicular environments (IEEE 802.11p).

In the second place, we will provide a brief introduction to H.264 and H.265, the video coding standards we used to compress and transmit the different video samples used for testing.

1.1.1 Wireless communications overview

A key factor for the success of the future Intelligent Transportation Systems is the compatibility between vehicles of different brands, road side units, and assistance vehicles. This compatibility is, of course, achieved by the usage of a communications standard.

1.1.1.1 IEEE 802.11

Since its presentation in the late 90's, the IEEE 802.11 standard has been the most widely used wireless local area network (WLAN) technology.

The standard introduced its basic Medium access Control (MAC) technique: the distributed coordination function (DCF).

DCF employs a Carrier sense multiple access with collision avoidance (CSMA/CA) mechanism to transmit the data packets. This mechanism is based on a very simple principle: listening to the medium before sending the packet. If the medium is clear for a certain time (DCF Interframe Space or DIFS), then a node might send its packet. In order to avoid collisions if two nodes attempt to send a packet, DCF also makes use of a binary exponential backoff algorithm to minimize collisions. So, when a node senses the medium clear for a DIFS time, it will initiate a random wait between zero and the value of the Contention Window (CW).

If during that period the medium remains clear, then the node sends the packet and waits for the acknowledgment (ACK). The acknowledgment has the highest priority, and it is sent after waiting for the medium to be clear for a SIFS (Short Interframe Space) period. The SIFS period is shorter than the DIFS.

If the node does not sense the ACK, it assumes that the sending has failed due to a collision, and restarts the process by increasing the contention window using the binary exponential backoff algorithm in order to minimize the probability of suffering a collision again.

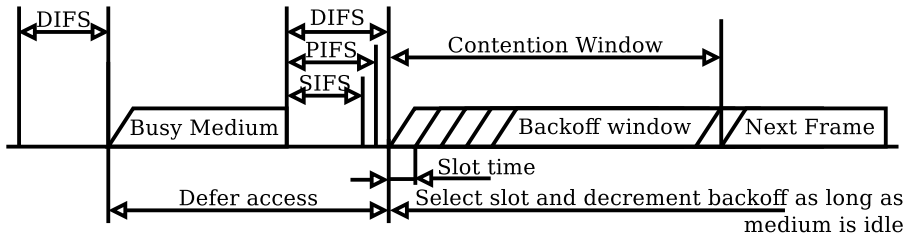


Figure 1.1: Detail of CSMA/CA mechanism.

Figure 1.1 shows a scheme of the CSMA/CA mechanism.

IEEE 802.11 defines different network structures. The main ones are BSS (Basic Service Set), and IBSS (Independent Basic Service Set).

The BSS mode is commonly referred as infrastructure mode. It requires a master node (commonly referred to as Access Point) to establish the network. Every other node connects to the AP and, therefore, every communication has to go through the AP.

The IBSS mode is commonly known as ad-hoc mode. The main advantage is that there is no requirement for a master node, and therefore nodes can establish spontaneous communication if they are within communication range.

1.1.1.2 IEEE 802.11e

To meet the need for MAC-level QoS, the IEEE 802.11e working group has enhanced the IEEE 802.11 standard to provide QoS at the MAC layer. MAC QoS is fundamental in contention-based wireless networks to achieve traffic differentiation in terms of both throughput and delay.

This amendment to the standard substitutes the DCF function with the new Enhanced distributed channel access (EDCA), which is backwards compatible with the original DCF function.

With EDCA, high-priority traffic has a higher probability of being sent than low-priority traffic: a station with high priority traffic waits less before sending its packets, on average, than a station with low priority traffic.

EDCA defines four traffic categories, Background (AC_BK), Best Effort (AC_BE), Video (AC_VI), and Voice (AC_VO). The differentiation and prioritization of the packets corresponding to each category when transmitting is achieved by modifying both the AIFS and the Contention Window for each category. Table 1.1, and figure 1.2 show the standard values for each access category.

Table 1.1: EDCA contention window values

Access Category	CW (min)	CW (max)	AIFS
Background	15	1023	7
Best Effort	15	1023	3
Video	7	15	2
Voice	3	7	2

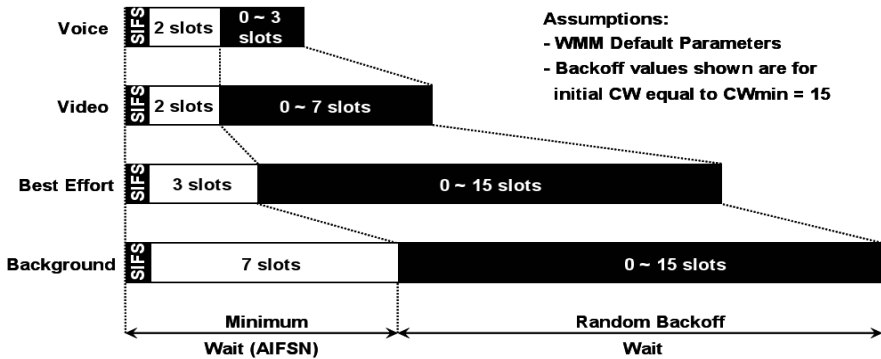


Figure 1.2: IFS, CWmin and CWmax for each traffic category.

1.1.1.3 IEEE 802.11p , DSRC, WAVE

Vehicular Ad-Hoc Networks have its own dedicated wireless standards. Basically, IEEE 802.11p is a mix of IEEE 802.11a with the IEEE 802.11e QoS extensions. That is, it makes use of the same basic technology as IEEE 802.11a, and includes as a default sending mechanism the IEEE 802.11e QoS extensions.

IEEE 802.11p makes use of the licensed Intelligent Transportation System (ITS) of 5.9 GHz (5.85-5.925 GHz), which provides a spectrum free of other “home devices”, and so specific devices are required to transmit in that frequency band.

As the communication link between two vehicles, or between a vehicle and an RSU, can be very short, IEEE 802.11p provides a mechanism to send and receive data frames without establishing a BSS, and thus without the slow steps of association and authentication. If association and authentication is required, they can be provided by upper layers.

Some agencies and studies sometimes refer to IEEE 802.11p as Dedicated short-range communications (DSRC), as it also refers

to the frequency band and lower layers for ITS communication. However, there are differences in DSRC regulations between Japan, Europe and U.S.A.

DSRC defines seven 10MHz channels with a transmission speed ranging from 6 to 54 Mbps. One of those channels is the control channel where RSUs offer their services, four channels are called “service channels” and can be used for whatever purpose, and finally the other two are special channels for public safety, one of them dedicated to intersections, and the other one is a High availability and low latency channel.

Wireless Access in Vehicular Environments (WAVE), which became part of the IEEE 1609 family of standards, defines an architecture and a complementary standardized set of protocols, services and interfaces that collectively enable secure vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) wireless communications.

Additionally to WAVE services, IEEE 802.11p also provides compatibility with the standard IPv6 network layer in the four service channels, but it does not offer compatibility in the safety and high availability channels.

1.1.2 Digital video coding techniques overview

Sending animated images to remote locations has been part of the human imaginary for centuries. The invention of television in 1926 by the Scotsman John Logie Baird led to the realization of this concept over analogue communication, but even though those systems made use of compression to fit the images into limited bandwidths, they were still analogue systems.

The emergence of mass-market digital video in the 1990s was a consequence of the efforts made in the 1970s to create efficient compression mechanisms. Even though the earliest digital video

formats were limited to very low resolution images (352x288 pixels or smaller) the continuous development and refinement of the different video compression mechanisms has allowed achieving an enormous video quality at low data rates.

Nowadays digital video has been adopted in almost every application that makes use of video, starting with cinemas, Digital Versatile Disks (DVD) and Bluray players, digital TV, etc. Even smartphones are being actively used for producing and recording digital video.

This adoption was made possible by the development of different video coding standards, which provide users with the ability to compress the video in a tiny space with a high quality. Standardization enables encoders and decoders from different manufacturers to work together across a range of applications.

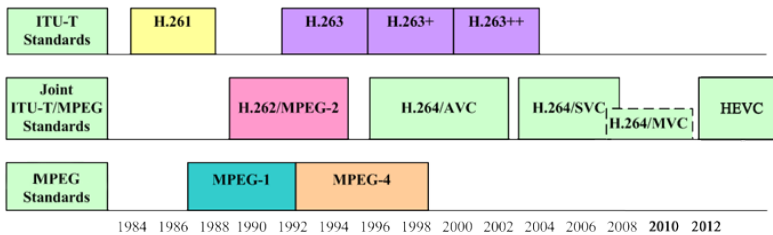


Figure 1.3: Evolution of the ITU-T and MPEG standards.

The creation of these standards is mostly being lead by two organizations, the ISO/IEC Motion Picture Experts Group (MPEG) and the ITU-T video coding experts group (ITU-VCEG). The ISO video coding standards are denoted with MPEG-x (MPEG-1, MPEG-2, ...), while the ITU-T ones are denoted with H.26x (H.261, H.262, ...). Some other organizations have published other video standards during these years such as (Theora, VP9, or VC-1), but these formats are not so widely extended as the MPEG and

ITU-VCEG standards.

For the past few years, MPEG and ITU-VCEG have been working together to create joint standards, thereby achieving a better compression by using technologies from groups of experts.

This section is meant to be an overview of the two video coding standards employed in this thesis: the Advanced Video Coding (H.264/AVC) and High Efficiency Video Coding (H.265/HEVC) standards.

1.1.2.1 H.264 (Advanced Video Coding)

H.264/AVC, also known as MPEG-4 part-10, is based on well-known and well-established techniques present in previous standards such as MPEG-2 and H.263. The final drafting work on the first version of the standard was completed in May 2003, and various extensions of its capabilities have been added in subsequent editions, like Multi View coding Extensions (MVC) that were added in 2009.

Coding process

The coding process is one of the inherited techniques, and consists of four stages.

1. Divide each frame into blocks of pixels in order to process the video at the block level.
2. Exploit the spatial redundancies present in the frame.
3. Exploit temporal dependencies between blocks in successive frames, that is, encode only the differences between successive frames.

4. Exploit the remaining spatial redundancies that may exist, as for example, encoding only the differences between original and predicted blocks.

Frame types

Inherited from previous standards, H.264/AVC can make use of different types of frames. Figure 1.4 illustrates the most used frame types I, P, and B frames.

An I frame, or intra frame, is a self-contained frame that can be independently decoded. The first image in a video sequence is always an I-frame, and these frames act as starting points if the transmitted bit stream is damaged.

The main drawback of this type of frames is that they consume a large amount of bytes as they do not exploit temporal redundancies.

A P frame, or predictive inter frame, references previous I or P frames to encode an image. P frames usually require less bits than I frames, but since they depend on previous frames, they are more susceptible to transmission errors and decoding artifacts.

Finally, a B frame, or bi-predictive inter frame, references both previous and future frames, thereby achieving a better compression, but being even more susceptible to transmission errors.

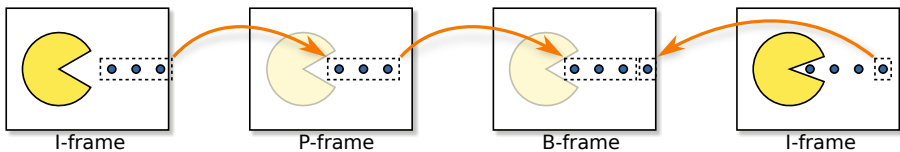


Figure 1.4: Types of frames. (Source: Wikimedia Commons)

Intra-prediction

H.264/AVC supports multiple directions to predict block displacements. As shown in figure 1.5, there are 9 different possibilities, covering the most common directions.

Intra-prediction is responsible of detecting spatial redundancies by comparing the 16x16 blocks within a frame.

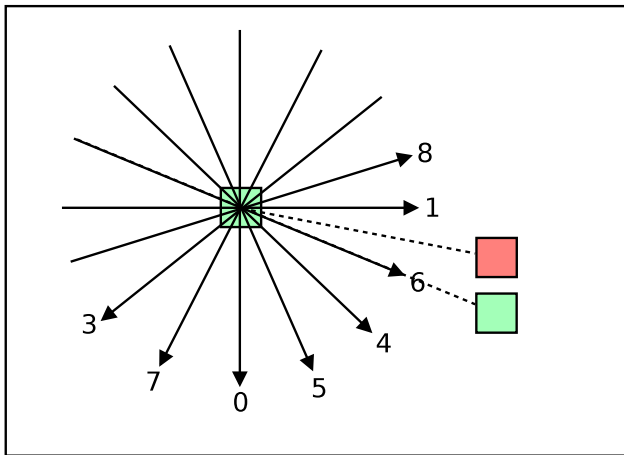


Figure 1.5: H.264 Intra Prediction.

Motion compensation

Motion compensation is the most demanding part of the encoding process as it searches temporal redundancies across frames of the video.

In H.264, 16x16 blocks can be divided into smaller blocks, as shown in figure 1.6. Macroblocks can be divided in four different sizes: 8x8, 8x4, 4x8, and 4x4.

Each of these block divisions must be compared with similar divisions in other frames, being thus a CPU intensive process.

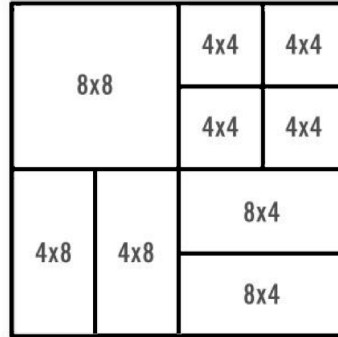


Figure 1.6: H.264 Macroblock

1.1.2.2 H.265 (High Efficiency Video Coding)

H.265/HEVC, also known as MPEG-H Part 2, is a new standard for video compression that has the potential to deliver better performance than earlier standards such as H.264/AVC.

The approach to video coding of the new HEVC is very similar to previous standards such as H.264 and MPEG-2, merely extending existing functionalities. The two main differences between H.264 and HEVC are increased modes for intra prediction and refined partitioning for inter prediction. This is mainly translated into more computing power required for encoding the video.

The new mode for intra prediction, which can be seen in figure 1.7, increases the possible directions from 9 to 35.

The most significant change referring to inter prediction is the size of the macroblock. Instead of macroblocks, H.265 pictures are

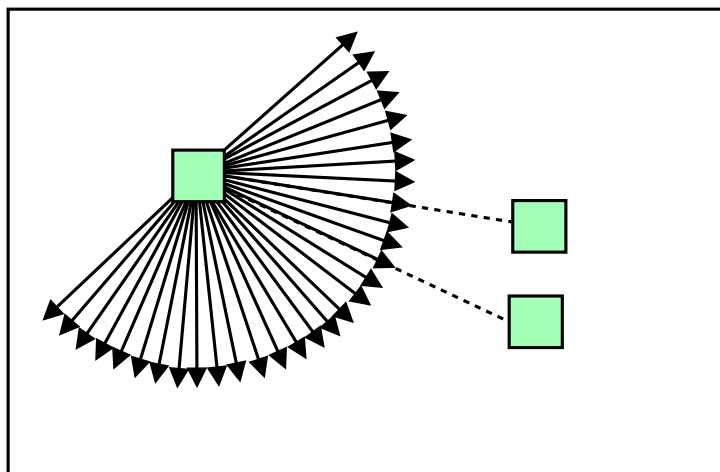


Figure 1.7: H.265 Intra Prediction

divided into so-called coding tree blocks, or CTBs for short. Depending on the stream parameters, they are either 64x64, 32x32 or 16x16. Each CTB can be split recursively in a quad-tree structure, all the way down to 8x8. So for example a 32x32 CTB can consist of three 16x16 and four 8x8 regions. These regions are called coding units, or CUs. CUs are the basic unit of prediction in HEVC. CUs can be 64x64, 32x32, 16x16 or 8x8. An example of this new CTB is shown in figure 1.8.

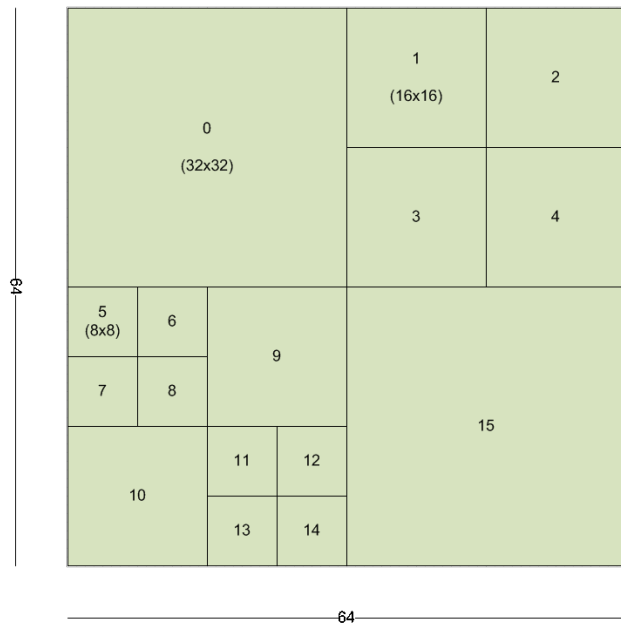


Figure 1.8: H.265 Coding Tree Block example.

1.1.2.3 Summary of differences

	H.264	H.265
Intra prediction	Multi-direction, multi-pattern, 9 intra modes for 4x4, 9 for 8x8, and 4 for 16x16	35 modes for intra prediction, 32x32, 16x16, 8x8 and 4x4 prediction size.
Coded Image Types	I, B, P, SI, SP	I, P, B
Transform	8x8 and 4x4 DCT-like Integer Transform	32x32, 16x16, 8x8 and 4x4 DCT-like Integer Transform
Motion Estimation Blocks	Up to 16x16. Can be divided into 16x8, 8x16, 8x8, 8x4, 4x8 and 4x4.	Coding Tree Units of 64x64 and hierarchical quad-tree partitioning down to 32x32, 16x16, 8x8 Each size can be partitioned even more.
Entropy Coding	Context adaptive binary arithmetic coding (CABAC) and context adaptive VLC tables (CAVLC)	Context adaptive binary arithmetic coding (CABAC)
Frame Distance for Prediction	Up to 16 past and/or future reference frames, including long-term references	Up to 15 past and/or future reference frames, including long-term references
Fractional Motion Estimation	1/2 pixel 6-tap filter, 1/4 pixel linear interpolation	1/4 pixel 8-tap filter
In-Loop Filter	Adaptive deblocking	Adaptive deblocking filter filter and sample adaptive offset filter

1.1.2.4 RTP packetization

H.264/AVC and H.265/HEVC streams can offer full RTP packetization. In fact H.264/AVC has its own RFC for RTP packetization, and a group of experts is working on the H.265/HEVC RFC [8].

Both video coding standards include a Network Adaptation Layer (NAL) whose main goal is to provide a “network-friendly” representation of the codec. The coded video data is organized into NAL Unit (NALU). Each NALU consists of a byte string of variable length that contains a 1 byte header to indicate the type of data that is contained in the NALU, and a payload containing the NALU data.

The format is defined in such a way that can be compatible with both RTP and Byte-Stream formats. For the Byte-Stream format a certain bit pattern is used to separate consecutive NALUs. When using RTP this pattern is removed, and the RTP separation is used in order to avoid redundant data.

The RTP packetization system is designed in such a way that permits both fragmenting a NALU into several RTP packets, and aggregating several NALUs in one RTP packet. These operations can be performed by all the media-aware network elements, since both codecs do not require processing the entire stream, the operation can be done by looking at the NALU header.

Furthermore, RTP packetization does not generate almost any overhead, and it allows detecting lost packets and distinguishing between essential and non-essential packets without having to decode the video stream.

1.2 Challenges

In the development of this doctoral work we had to face several challenges, as the network characteristics are very particular. These challenges can be summarized as follows:

Limited resources

Wireless communications suffer from limited spectrum resources. Since every vehicle should be connected to the same channel to achieve a full connectivity, it is quite common that messages are lost whenever two vehicles try to communicate at the same time; this problem is frequent on both unicast and multicast scenarios. Furthermore, since the WAVE standard considers the availability of several communication channels, vehicles may have to switch between the control channel and the service channel, thereby halving the effective bandwidth.

High mobility

Mobile ad-hoc networks have to cope with a certain degree of mobility, which creates connectivity and path loss problems. Such problems are aggravated in vehicular ad-hoc networks due to the relative high speeds involved. In highway scenarios vehicles only stay within the range of transmission of a road side unit (RSU) for a few seconds. In unicast scenarios it is necessary to obtain and maintain a route between the two communicating nodes. This requirement is almost impossible to meet if we think of two random vehicles on the road, but instead, if we think of a group of vehicles moving towards the same destination (group of firefighters, ambulances or police), they can form a platoon where the relative speeds are lower, thus simplifying the routing problem.

Decentralized system

Since every vehicle is independent from others, the developed systems should be fully decentralized. Decentralized systems have to cope with several problems, being the absence of a coordinator the main one. Since the type of networks employed in this thesis share the same channel, a good channel access regulation is required in order to maximize the throughput and minimize the delay.

Performance measurement

Measuring the performance of the different proposed solutions is not an easy task, especially in real-world scenarios. In these scenarios it is easy to measure throughput as it only requires counting the received data. On the other hand, accurate end-to-end delay results are hard to achieve as a result of clock desynchronization and clock drift between two different computers. In addition, since we are working with video content, not only throughput and delay are important, but also stability of the data rate and final video quality.

1.3 Methodology

In this thesis we employed a novel methodology compared to previous works.

First we reviewed the state of the art in order to find the best contributions in the field of study.

Second, and contrarily to other previous studies, we focused our efforts on a real implementation of one of the algorithms previously studied when reviewing the state of the art. Thanks to this real

world implementation we were able to obtain information about the simulations' accuracy and precision, as well as an overview of the different problems associated with real world implementations and limitations. Thanks to this previous experience, the proposals of this thesis are easy to implement in a real-world scenario without having the need of modifying existing standards.

The last step of this thesis consisted in the proposal and evaluation of several real-time video flooding schemes for vehicular ad-hoc networks. The different proposals were evaluated in several simulated scenarios. During the design process of the different solutions we always had in mind the characteristics of real environments experienced previously.

In order to increase the accuracy of the results, the simulated scenarios were as close as possible to a real world scenario. Additionally, we obtained video samples to verify the real quality of the transmitted videos.

1.4 Objectives

This thesis aims at evaluating the ability of transmitting real-time video in vehicular networks, offering a set of solutions for both unicast and multicast video.

In order to achieve the main goal of the thesis, a set of objectives has been proposed:

1. Evaluate the accuracy of the simulations in the field of mobile ad-hoc networks, in terms of throughput, latency, and general behavior of typical protocols.
2. Measure the actual behavior of the IEEE 802.11e QoS mechanism for traffic differentiation when applied to real multihop ad-hoc networks.

3. Create solutions for both, unicast and multicast scenarios to cover almost every possible usage, present and future.
4. Maintain compatibility with current and future standards.
5. Create a flooding algorithm efficient enough to allow real-time video transmission in long distances.

1.5 Organization of the thesis

This thesis is presented by a compilation of articles, and so, the rest of the thesis is organized as follows:

Chapters 2 to 8 include the different publications relevant to this thesis adapted to the required format.

Chapter 9 includes a discussion of the results obtained where the most relevant contributions to the field are summarized.

Chapter 10 presents the conclusions of this work, as well as some ideas for future work.

Finally, chapter 11 includes a summary of the different publications presented in this thesis.

Publication A:
**Deploying a real IEEE 802.11e
testbed to validate simulation results**

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Deploying a real IEEE 802.11e testbed to validate simulation results*. IEEE 34th Conference on Local Computer Networks. (LCN 2009). Pages 109 to 115, October 2009.

Abstract

The IEEE 802.11e technology is receiving much interest due to the enhancements offered to wireless local area networks in terms of QoS. Other application fields for this technology are wireless ad-hoc networks, wireless mesh networks and vehicular ad-hoc networks. In the literature, most of the research works available focusing on the IEEE 802.11e technology offer simulation results alone,

being hard to find empirical results of real implementations. Additionally, we consider that the IEEE 802.11e implementation on simulation platforms has not been thoroughly validated using real-life results. In this work we analyze the performance of the IEEE 802.11e technology in a real multi-hop ad-hoc network testbed, comparing the results obtained with those of the ns-2 simulation platform. Experimental results show a significant consistency in terms of overall trends, although remarkable differences can be appreciated in terms of both delay and throughput results. In general, we find that simulation-based results are always optimistic compared to real testbed performance.

2.1 Introduction

Wireless Ad-Hoc Networks are packet radio networks composed by independent and heterogeneous stations that cooperate in routing and packet forwarding tasks, conforming a dynamic multi-hop network.

Interest in these networks has been growing in the last few years since they offer an adequate solution for military and disaster relief scenarios, ubiquitous computing environments, as well as to extend the coverage of networks mainly to provide Internet access to members of large disperse communities (e.g. university campus, rural areas, etc.)

The proliferation of devices with multimedia and wireless networking capabilities pave the way towards ubiquitous audiovisual communication among peers. Thus, an increasing interest in supporting QoS in ad-hoc networks has arisen. To meet this need, the IEEE 802.11e [27] working group has enhanced the IEEE 802.11 standard [26] to provide QoS at the MAC level. MAC level QoS is fundamental in contention-based wireless networks to achieve

traffic differentiation in terms of both throughput and delay. In wireless ad-hoc networks this requirement gains further significance since the complex radio interference issues associated with multi-hop channel access cause performance to be poor.

In research works such as [19, 50] authors offer an analysis of IEEE 802.11e in environments where stations communicate with an access point. Other works such as [16] and [23] study the performance of the IEEE 802.11e technology in infrastructureless ad-hoc networks, both static and mobile. The results presented in works such as the ones referred above are usually based on simulation or analytic models. While such approach is interesting for a first assessment of the performance and viability of QoS solutions, they tend to be optimistic and miss several problems associated with real-life implementations, such as hardware limitations, background noise and interferences, software restrictions and, last but not the least, bugs in the wireless cards' driver.

In this paper we setup an IEEE 802.11e enabled testbed using native Linux drivers that allows us to assess the performance of this technology in multi-hop ad-hoc network environments. We also validate the simulation results obtained with the ns-2 simulator, comparing them to the ones obtained in the testbed. Experiments evidence some significant performance differences in terms of throughput and delay.

The structure of this paper is the following: in the next section we refer to some related works regarding IEEE 802.11e testbeds. In section 2.3 we briefly introduce the IEEE 802.11 technology, as well as the enhancements proposed in the current IEEE 802.11e standard to support QoS. In section 2.4 we describe the characteristics of our testbed in terms of both hardware and software configurations. Experimental results obtained with both the testbed and the ns-2 simulator are presented in section 2.5. Finally, section 2.6 concludes the paper.

2.2 Related Works

In the literature we can find a few recent works where the authors setup a testbed to evaluate the performance of the IEEE 802.11e technology. In [38] Ng et al. setup a testbed to demonstrate that the 802.11e parameters CWmin, TXOP and AIFS work largely in line with analytic and simulation predictions. Additionally, they use the testbed to demonstrate some known problems with TCP's performance caused by cross-layer interaction between the TCP congestion control algorithm and the MAC layer CSMA/CA contention mechanism, as well a possible solution to that problem.

Dangereld et al. [21] have studied the delay in the context of protecting a voice call competing against data traffic. In particular, they present experimental measurements which demonstrate that the use of 802.11e offers a practical solution to successfully deliver quality of service to voice traffic in a mixed voice/data environment.

Banchs et al. [13] focus on applications of EDCA, i.e., the *enhanced distributed channel access* proposed by the IEEE 802.11e standard. In particular, the target applications are traffic engineering and service guarantees. Experimental results show that EDCA can effectively be used for traffic engineering purposes. However, the goal of providing service guarantees with EDCA is shown to be more challenging.

Vasilos and Stamatakis [46] investigate the optimal selection of minimum contention window values to achieve proportional fairness in a multirate IEEE 802.11e test-bed. The test-bed experiments show that when stations have different transmission rates and the same weight, proportional fairness achieves higher performance than the time-based fairness approaches, in terms of both aggregate utility and throughput.

All of the aforementioned testbeds were configured for the infrastructure mode alone, using Atheros wireless cards (either PCI or PCMCIA) and the MadWifi driver [4] for testing. Notice that this driver does not fully support the IBSS mode of operation, which has hindered researchers from testing the benefits of the IEEE 802.11e technology in ad-hoc network environments. Additionally, the driver does not support USB devices nor most of the wireless cards integrated in laptops.

To the best of our knowledge, our work is pioneer in terms of performance assessment of the IEEE 802.11e technology in a multi-hop ad-hoc network environment. Moreover, we rely on enhanced versions of standard wireless drivers for the Linux kernel that support state-of-the-art USB and integrated wireless cards, which favors a widespread adoption of this technology among both researchers and regular users.

2.3 Overview of the IEEE 802.11 Standard

The IEEE 802.11 standard [26] is a technology whose purpose is to provide wireless access to local area networks. Stations using this technology access the wireless medium using either the Point Coordination Function (PCF) or the Distributed Coordination Function (DCF).

The Point Coordination Function is a centralized access mode optionally used in a Basic Service Set (BSS) when a point coordinator (PC) is available. The PC is typically an *access point* (AP), and so the stations are said to operate in infrastructure mode. When relying on the PCF, contention-free periods (CFP) and contention periods (CP) alternate over time. The regular generation of beacons

allows stations to associate and synchronize with the PC. Typically a CFP is started after a beacon management frame, followed by a CP; together they conform a superframe. During the CFP there is no contention, meaning that stations are merely polled by the PC. The CF-End control frame is transmitted by the PC to indicate the end of the CF period, and the beginning of the CP. During the CP stations access the medium using the DCF.

The Distributed Coordination Function (DCF) uses a listen-before-talk scheme named *carrier sense multiple access* (CSMA) with *collision avoidance* (CA). It is used by stations in a BSS during the CP and also by stations in an IBSS operating in ad-hoc mode. The CSMA technology distributes the medium access task among all stations, making every station responsible for assuring the delivery of *MAC service data units* and reacting to collisions. The collision avoidance (CA) scheme is used to reduce the probability of collisions between different stations. To achieve this it applies a backoff procedure before initiating a transmission if the medium was not found to be previously idle. Stations select a random number of slots to wait before transmission in the interval between 0 and the current *contention window* (CW) value. The value for CW is set initially to the minimum value for the radio technology being used (CW_{min}), being increased when consecutive collisions occur up to a maximum value (CW_{max}).

The CSMA/CA mechanism shows good adaptation to different numbers of transmitters, and probabilistically shares the channel equally among all competing stations. However, it offers no mechanisms to perform traffic differentiation, making QoS support practically unfeasible. The IEEE 802.11e working group was created to add QoS support to the original IEEE 802.11 standard, and in 2005 a new international standard [27] was released.

2.3.1 IEEE 802.11e: MAC enhancements for QoS

The IEEE 802.11e working group has extended the IEEE 802.11 MAC in order to provide QoS support. This standard introduces the *hybrid coordination function* (HCF) which defines two new medium access mechanisms to replace PCF and DCF. These are the *HCF controlled channel access* (HCCA) and the *enhanced distributed channel access* (EDCA).

With the HCF there may still exist a contention period and a contention-free period in a superframe, but now the HCCA is used in both periods, while the EDCA is used only during the CP. This new characteristic of HCF obviates the need for a CFP since it no longer depends on it to provide QoS guarantees.

With 802.11e, the *point coordinator* is replaced by a *hybrid coordinator* (HC) which also resides in an AP. A BSS including a HC is referred to as a QBSS. In this paper we focus on ad-hoc networks and, therefore, we are only interested in 802.11e stations implementing EDCA. For more information on HCs, the HCF and the HCCA refer to [27].

Concerning 802.11e enabled stations forming an ad-hoc network, these must implement the EDCA algorithm. The 802.11e QoS support is achieved through the introduction of different *access categories* (ACs), and their associated backoff entities.

Contrarily to the legacy IEEE 802.11 stations, where all packets have the same priority and are assigned to a single backoff entity, IEEE 802.11e stations have four backoff entities (one for each AC) so that packets are sorted according to their priority. The different access categories available in IEEE 802.11e stations are: Voice (AC_VO), Video (AC_VI), Best-effort (AC_BE) and Background (AC_BK).

Each backoff entity has an independent packet queue assigned to it,

as well as a different parameter set. In IEEE 802.11 legacy stations, this parameter set was fixed, and so the inter-frame space was set to DIFS and the CWmin and CWmax were set to 15 and 1023 respectively (for IEEE 802.11a/g). With IEEE 802.11e the inter-frame space is arbitrary and depends on the access category itself (AIFS[AC]). We also have AC-dependent minimum and maximum values for the contention window (CWmin[AC] and CWmax[AC]). Additionally, IEEE 802.11e introduces an important new feature referred to as transmission opportunity (TXOP). A TXOP is defined by a start time and a duration; during this time interval a station can deliver multiple frames consecutively without contention with other stations. This mechanism, also known as *contention-free bursting* (CFB), increases global throughput through a higher channel occupation. An EDCA-TXOP (in contrast to an HCCA-TXOP) is limited by the value of TXOPLimit, which is a parameter defined for the entire QBSS and that also depends on the AC (TXOPLimit[AC]).

Table 2.1 presents the default MAC parameter values for the different ACs. Notice that smaller values for the AIFSN, CWmin and CWmax parameters result in a higher priority when accessing the channel; relative to the TXOPLimit, higher values result in larger shares of capacity and, therefore, higher priority.

Table 2.1: IEEE 802.11e MAC parameter values for a IEEE 802.11a/g radio

<i>Access category</i>	<i>AIFSN</i>	<i>CW_{min}</i>	<i>CW_{max}</i>	<i>TXOPLimit (ms)</i>
AC_BK	7	15	1023	0
AC_BE	3	15	1023	0
AC_VI	2	7	15	3.008
AC_VO	2	3	7	1.504

The relation between $AIFS[AC]$ and $AIFSN[AC]$, is the following:

$$AIFS[AC] = SIFS + AIFSN[AC] \times aSlotTime \quad (2.1)$$

where $SIFS$ is the shortest inter-frame space possible and $aSlotTime$ is the duration of a slot. As defined by the standard, the $AIFSN[AC]$ parameter must never be less than 2 to avoid interference with normal AP operation.

For applications to take advantage of the IEEE 802.11e technology, datagrams should have their IP TOS header field set according to the desired user priority. When delivered to an IEEE 802.11e enabled wireless card driver, those datagrams will be handled according to the priority defined. The mapping between user priorities (network layer) and the four medium access categories defined by the IEEE 802.11e standard (MAC layer) is shown in Table 2.2 for reference.

Table 2.2: User Priority to IEEE 802.11e Access Category Mapping (according to IEEE 802.1D)

User Priority	Designation	Access Category
1	BK (Background)	AC_BK
2	BK (Background)	AC_BK
0	BE (Best-effort)	AC_BE
3	EE (Video/Excellent-effort)	AC_BE
4	CL (Video/Controlled Load)	AC_VI
5	VI (Video)	AC_VI
6	VO (Voice)	AC_VO
7	NC (Network Control)	AC_VO

2.4 Testbed Setup

Figure 2.1 shows the multi-hop ad-hoc network scenario we have selected for our experiments. The scenario includes four source-destination pairs, as well as three intermediate stations, thus conforming a total of 11 stations. Unless stated otherwise, all traffic sources are set to generate the same data rate in all four ACs, and the traffic type chosen is constant bitrate (CBR) with packet size fixed at 512 bytes to fully control the load being injected. In our experiments we avoid synchronization among traffic sources to prevent results from being biased towards a worst case scenario; this was achieved by adding a small jitter to the inter-packet interval.

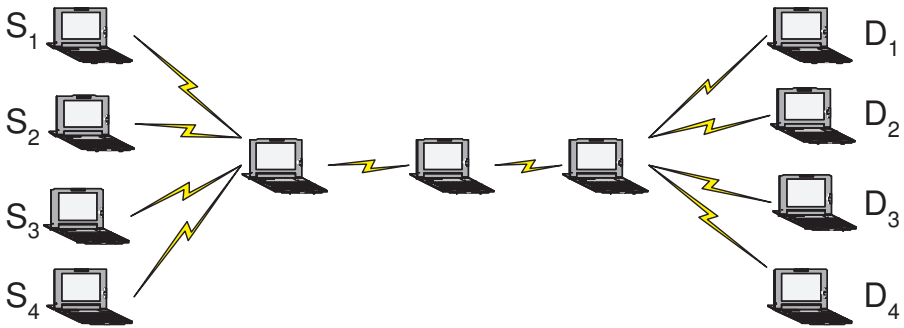


Figure 2.1: Multi-hop ad-hoc network scenario used for testing.

The simulation results used for comparison were obtained using the ns-2 simulator [30]. It was configured with the same scenario, and all nodes have an IEEE 802.11g radio operating at a fixed data rate of 54 Mbit/s. The distance between stations that are one hop away is of 200 meters, and simulation takes place within an area sized 1000×400 meters. Notice that the carrier sensing range is of 550 meters, meaning that interferences make spatial reuse to be significantly reduced (in fact, it is null for the center station).

For our testbed we could not use exactly the same settings due to space and radio range restrictions. In fact, to operate efficiently with IEEE 802.11g cards at a fixed data rate of 54 Mbit/s, stations must be at most a few meters away. Thus, in our testbed all the stations are within the same room (separated by about 1 meter), and the topology shown in Figure 2.1 was enforced logically through an adequate configuration of the firewall using the *iptables* tool [9]. Despite the total channel capacity of the testbed is reduced with respect to the simulator, we consider that it offers the best trade-off when assuming basic feasibility requirements. Nevertheless, when performing result comparisons, we shall take this restriction into consideration.

Due to the presence of other sources of noise in the 2.4 GHz band, we used the Wi-Spy spectrum analyzer [5] to aid us in determining the best channel for experimentation. The characteristics of the channel used are shown in Table 2.3 for the sake of completeness. Those values were obtained with all the stations participating in the testbed turned off.

Table 2.3: Characteristics of the wireless channel selected (obtained with Wi-Spy 2.4GHz spectrum analyzer).

<i>Parameter</i>	<i>Value range</i>
Average Floor (dBm)	-85.94
Average Peak (dBm)	-79.13
Duty Cycle	11.06

With respect to the IEEE 802.11e technology, the stations used in our testbed adopt the EDCA channel access parameters shown in Table 2.1, and the contention-free bursting (CFB) mechanism is turned off. The same applies to virtual stations participating in the ns-2 based simulation experiments. Notice that ns-2 does not currently include support for IEEE 802.11e, which required

installing the extensions developed by Wietholter and Hoene [48].

2.4.1 Hardware and software details

To create the multi-hop network presented in Figure 2.1 we used 11 computers, where 8 of them were low-cost Asus EeePC 901 netbooks, 1 was a high performance Lenovo laptop, and the remaining 2 were fully equipped desktop PCs.

In terms of wireless cards, we opted for using IEEE 802.11e enabled cards using the Ralink chipset [7]. Our decision was based on the availability of the source code for the Ralink wireless card drivers for Linux, a basic requirement to allow performing the required enhancements. In particular, on the Asus netbooks we used their integrated wireless card (Ralink RT 2860 chipset), and on the remaining terminals we used either D-Link DWA-140 or Linksys WUSB600N-EU wireless USB cards, both based on the Ralink RT 2870 chipset.

All the terminals involved in the testbed had a GNU/Linux operating system installed, either version 2.6.27 (netbooks and laptop) or 2.6.24 (desktop). The Ralink wireless card drivers used were 1.7.0.0 for the RT2860 chipset, and 1.3.1.0 for the RT2870 chipset¹.

The main problem encountered when attempting to deploy an infrastructureless wireless network using the aforementioned hardware was the lack of support for IEEE 802.11e in the ad-hoc mode of operation. Such setback evidences the lack of interest shown by hardware manufacturers in this issue, requiring significant enhancements to the original drivers in order to make EDCA channel access operative in the infrastructureless mode. Additionally, further enhancements were made to add support for the 2.6.27 Linux

¹These drivers are available at:
<http://www.ralinktech.com/ralink/Home/Support/Linux.html>

kernel, to allow fixing the data rate at 54 Mbit/s (a basic requirement to make results comparable with those of ns-2), as well as to correct some bugs detected.²

2.4.2 Throughput and delay measurements

While simulation traces allow to easily obtain all required metrics, in a real testbed the process is more complex. In particular, measuring the throughput required enabling the *tcpdump* capture tool, based on *libpcap* [53], on all traffic destinations (stations D_i). The captured traffic was then filtered to discriminate throughput among the different QoS categories; finally, the values provided by the different destinations were aggregated to obtain per-category throughput values.

Concerning delay, the measurement process is significantly more complex since sender/receiver clocks are not synchronized. Different delay measurement possibilities were discussed in [21], and for this testbed we adopted the solution proposed in [18], where a Fast Ethernet backbone is used to forward traffic back to the source, thus allowing to compute delay using a same clock. To make these measurements straightforward we developed our own UDP traffic generator. This traffic generator includes a time-stamp at the beginning of the packet's payload and, when receiving it again through the feedback channel, clock differences allow to compute delay.

To make sure that the Fast Ethernet return path did not affect delay measurements in a significant manner we calculated the additional delay introduced by the wired channel for different load values. The experimental results are shown in Figure 2.2.

²Drivers are available upon request.

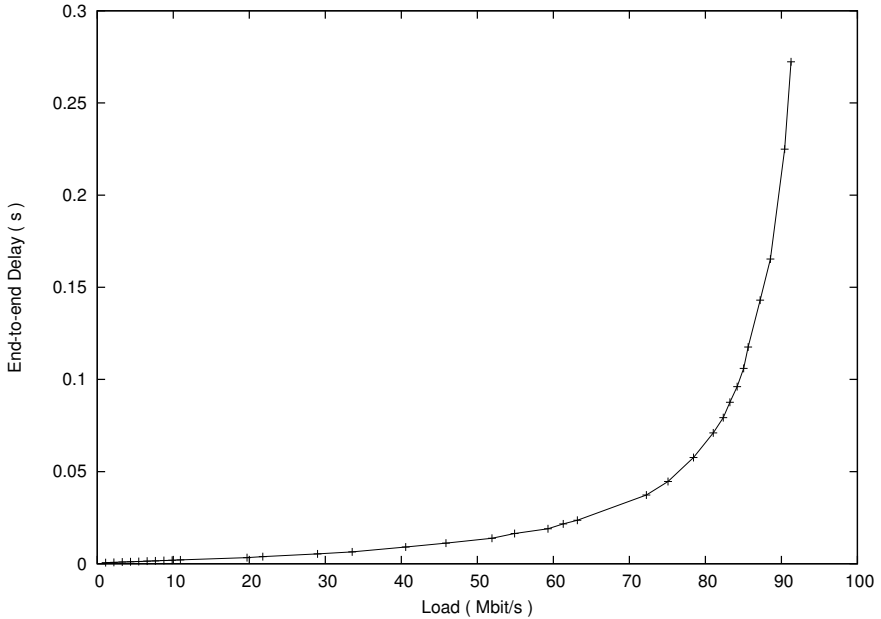


Figure 2.2: Load vs. delay values for the Fast-Ethernet link used as return path.

The maximum load measured on the return channel during our testbed experiments never surpassed 5 Mbit/s, which means that the additional delay introduced by the Fast-Ethernet link is rather small. Nevertheless, we used these results to obtain unbiased estimations of all delay values, thus measuring the delay associated with the wireless counterpart much more precisely.

2.5 Experimental Results

In this section we present a performance assessment study of the IEEE 802.11e technology in a multi-hop ad-hoc network environ-

ment. We run the same set of tests using both the wireless testbed described in the previous section and the ns-2 simulator. All experiments run for 120 seconds, being that the first and the last 10 seconds of each experiment were discarded. The chosen performance metrics are the throughput and the end-to-end delay values. Each experiment was repeated 20 times, which allowed us to determine, with a degree of confidence of 95%, that the maximum error for the mean is under 5% for all the values presented in this section.

We have split our analysis into two sets of experiments. In the first set of experiments our purpose is to assess the traffic differentiation capabilities of the EDCA channel access mechanism in the proposed multi-hop ad-hoc network. In the second set of experiments our goal is to determine the degree of robustness and stability that the two highest access categories (Video and Voice) experience when varying the amount of best effort and background traffic.

2.5.1 Traffic differentiation analysis

When a network is very lightly loaded QoS mechanisms are both ineffective and unnecessary. However, as we increase traffic to moderate/high loads, QoS mechanisms become fundamental to discriminate among the different types of traffic. Therefore, in the current set of experiments we gradually increase the traffic load by the same amount for all four MAC categories defined by the IEEE 802.11e standard.

Figure 2.3 shows the effectiveness of the IEEE 802.11e technology in terms of traffic differentiation at different loads. Comparing the results obtained with the simulator with those obtained in our testbed allows us to draw several conclusions: (i) in a real

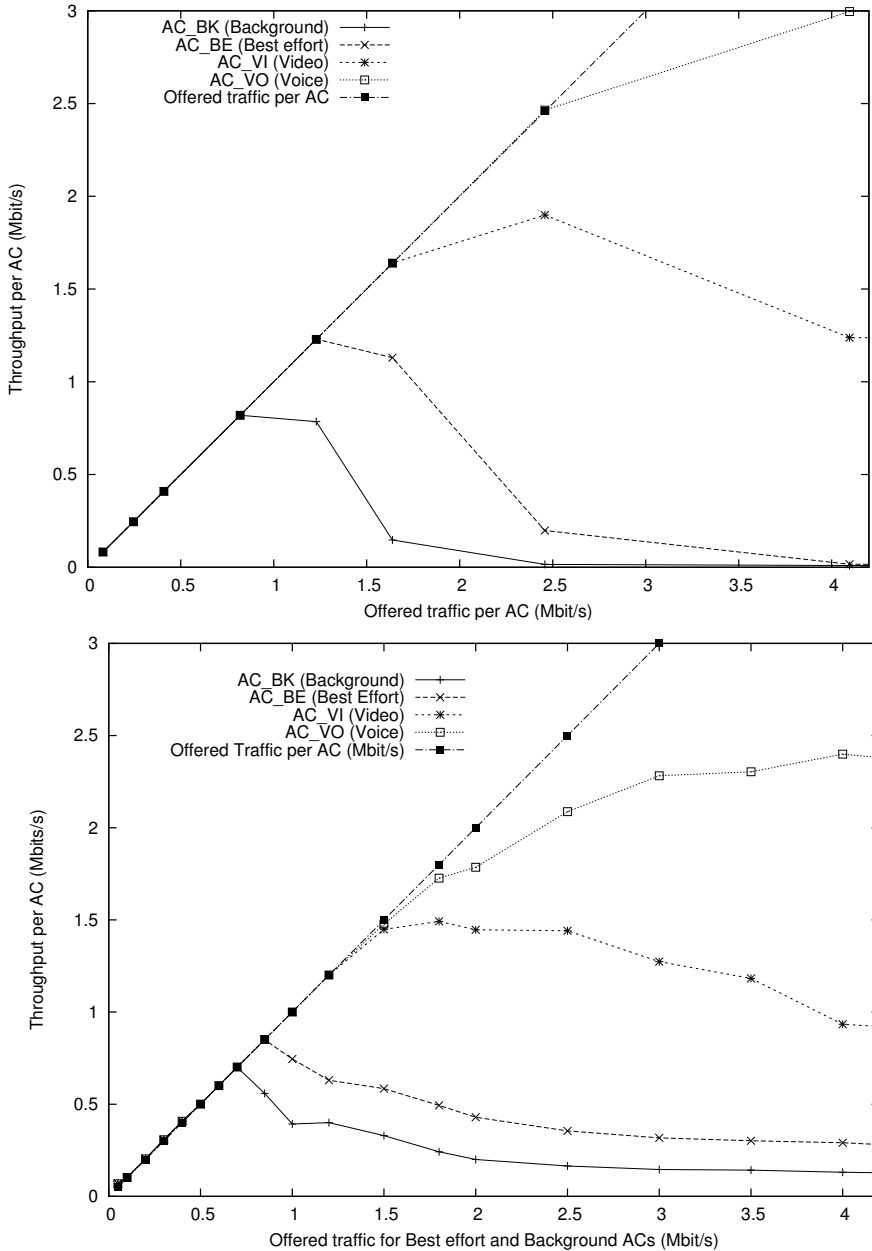


Figure 2.3: Throughput values achieved via simulation (top) and in our testbed (bottom).

2.5 Experimental Results

testbed the throughput values achieved before reaching the saturation points do not always present a matching with the offered load due to more frequent losses; this is especially noticeable for the voice and video traffic categories, and is typically associated with channel noise and external sources of interference; (ii) in the testbed the throughput for all traffic types decays more abruptly when the saturation point for each access category is reached; additionally, we also observe that the particular saturation point for each access category is reduced compared to simulation results. This effect is related to the lower channel capacity of the testbed environment due to the overlapping in terms of radio coverage and the presence of beacons; (iii) the traffic differentiation effectiveness of IEEE 802.11e is partially lost in the testbed compared to the simulation case, which allows to avoid starvation of best effort and background traffic by reducing QoS traffic (mostly in the video access category).

Figure 2.4 shows the results for this same set of experiments, but now focusing on the end-to-end delay metric instead. As expected, the most significant performance issues referred above guard a strict relationship with the end-to-end performance. In particular, we should remark that: (i) contrarily to the simulation experiments, the delay values obtained in the testbed experience a much more steep growth for low load values (notice that a logarithmic scale is used for the vertical axis); and (ii) when reaching moderate/high saturation levels the differentiation capabilities of IEEE 802.11e in the testbed experience a significant loss of effectiveness compared to simulation results. In particular, average delay values for video and voice traffic are increased by one order of magnitude; such difference should be considered quite significant and taken into consideration when designing QoS architectures based on the IEEE 802.11e technology.

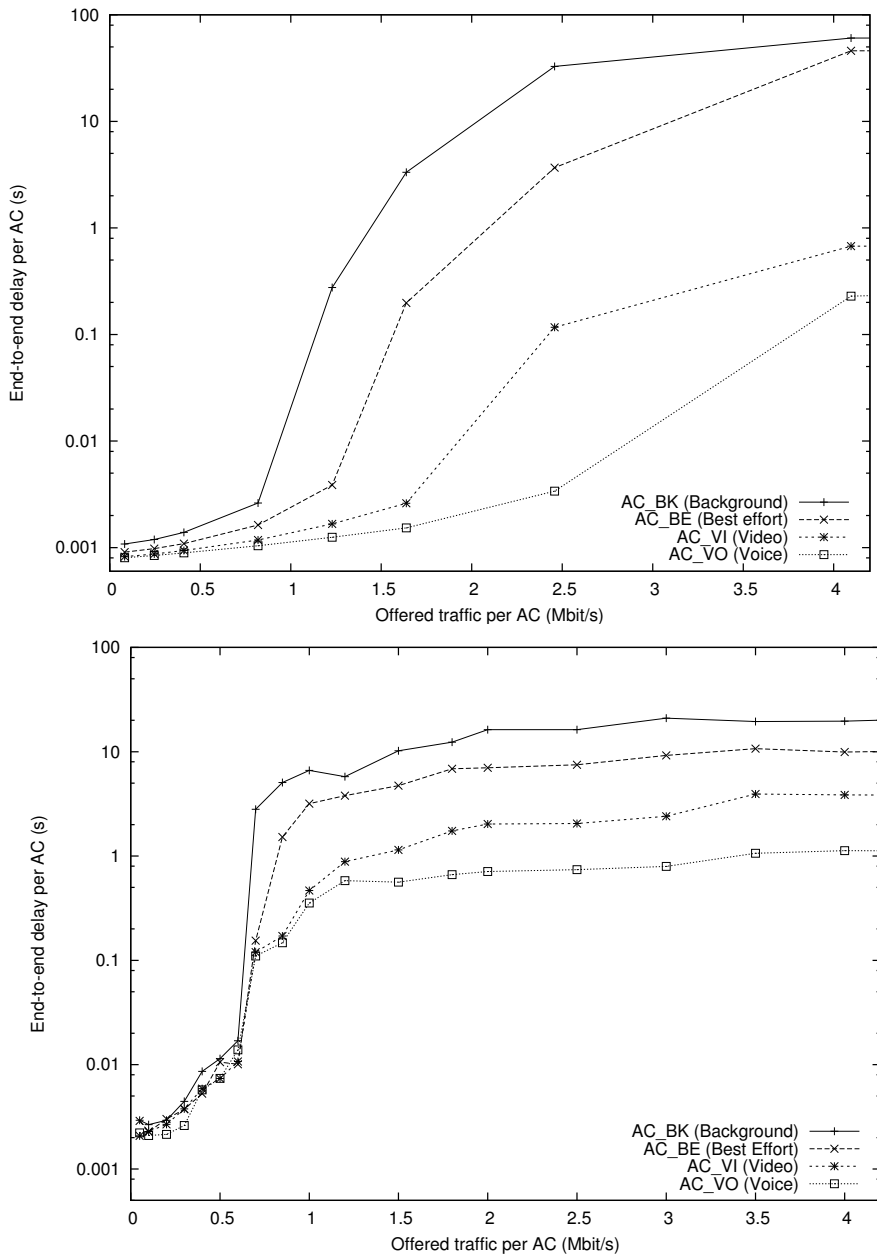


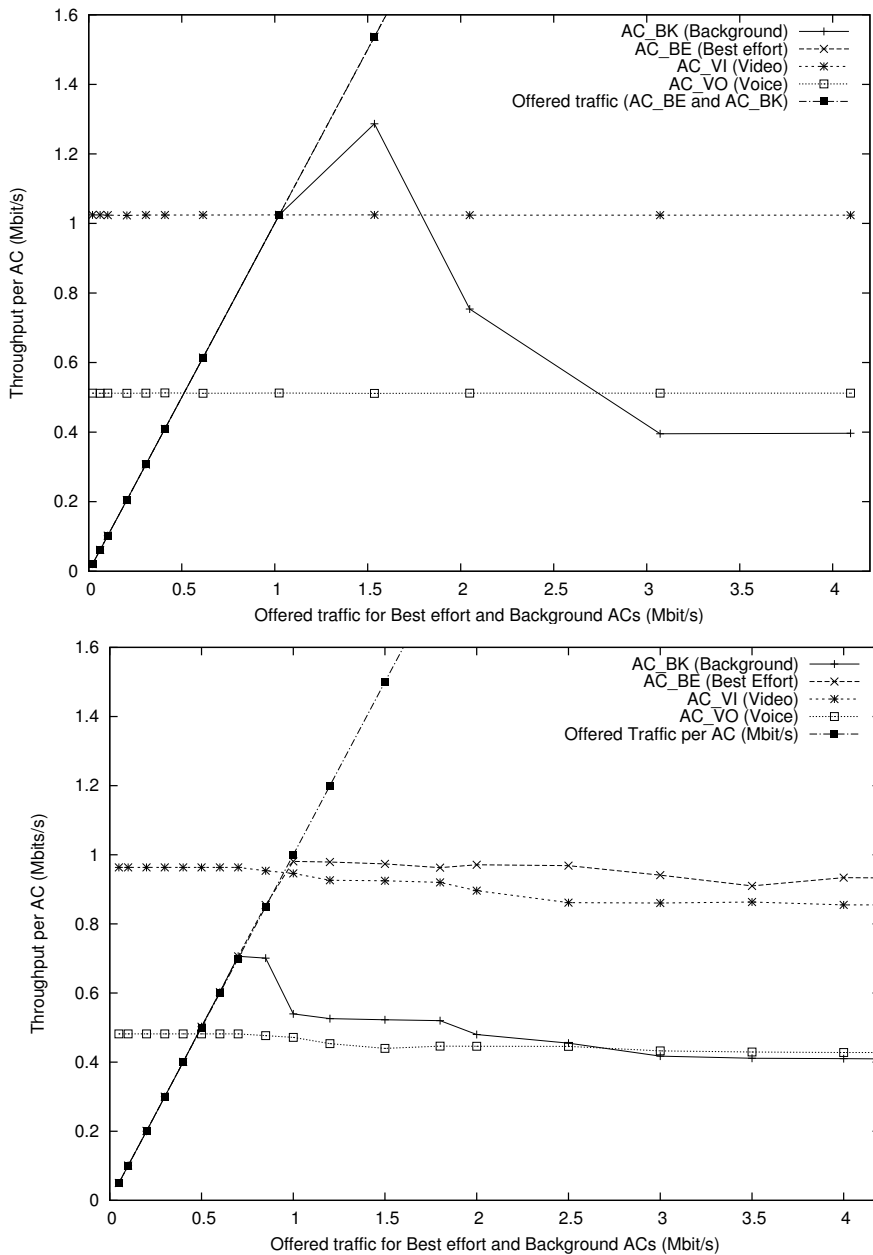
Figure 2.4: End-to-end delay values achieved via simulation (top) and in our testbed (bottom).

2.5.2 QoS robustness analysis

We now present the results related to the second set of experiments, where the aim is to assess whether the IEEE 802.11e technology is able to offer steady performance levels to active QoS connections when increasing non-QoS traffic. With this purpose we fix the aggregated load for the video and voice traffic at 1 and 0.5 Mbit/s, respectively, increasing load levels for the best-effort and background traffic categories alone.

Figure 2.5 shows the throughput results obtained using the wireless testbed and the ns-2 simulator. It is noticeable that in the testbed the throughput values for video and voice traffic are not maintained stable, experiencing packet losses of up to 11.5% and 11.8%, respectively. This is in clear contrast with simulation results, where throughput levels are maintained practically unaltered for all load values. As a secondary issue, we noticed that the throughput values for best effort traffic under saturation differ greatly, leveling out at 2.4 Mbit/s in the ns-2 simulator (not visible due to scale), while in the testbed they level-out at about 0.9 Mbit/s. The observed differences, are mainly due to the lower overall channel capacity of the testbed environment, as explained earlier.

In terms of end-to-end delay, the results presented in Figure 2.6 guard some resemblance with the ones of Figure 2.4, and again evidence the difficulties at maintaining reduced delay values for video and voice traffic in a real testbed. The most noticeable difference is detected in the testbed results at load values close to 0.7 Mbit/s, where a sharp increase in the delay values of all traffic categories occurs. In particular, video and voice traffic experience an increase of about two orders of magnitude compared with simulation results, while the increase for both best effort and background traffic is limited to one order of magnitude. Thus, the high degree of delay stability obtained via simulation is lost. In terms of ab-



66
 Figure 2.5: Throughput stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (top) and from the testbed (bottom).

2.5 Experimental Results

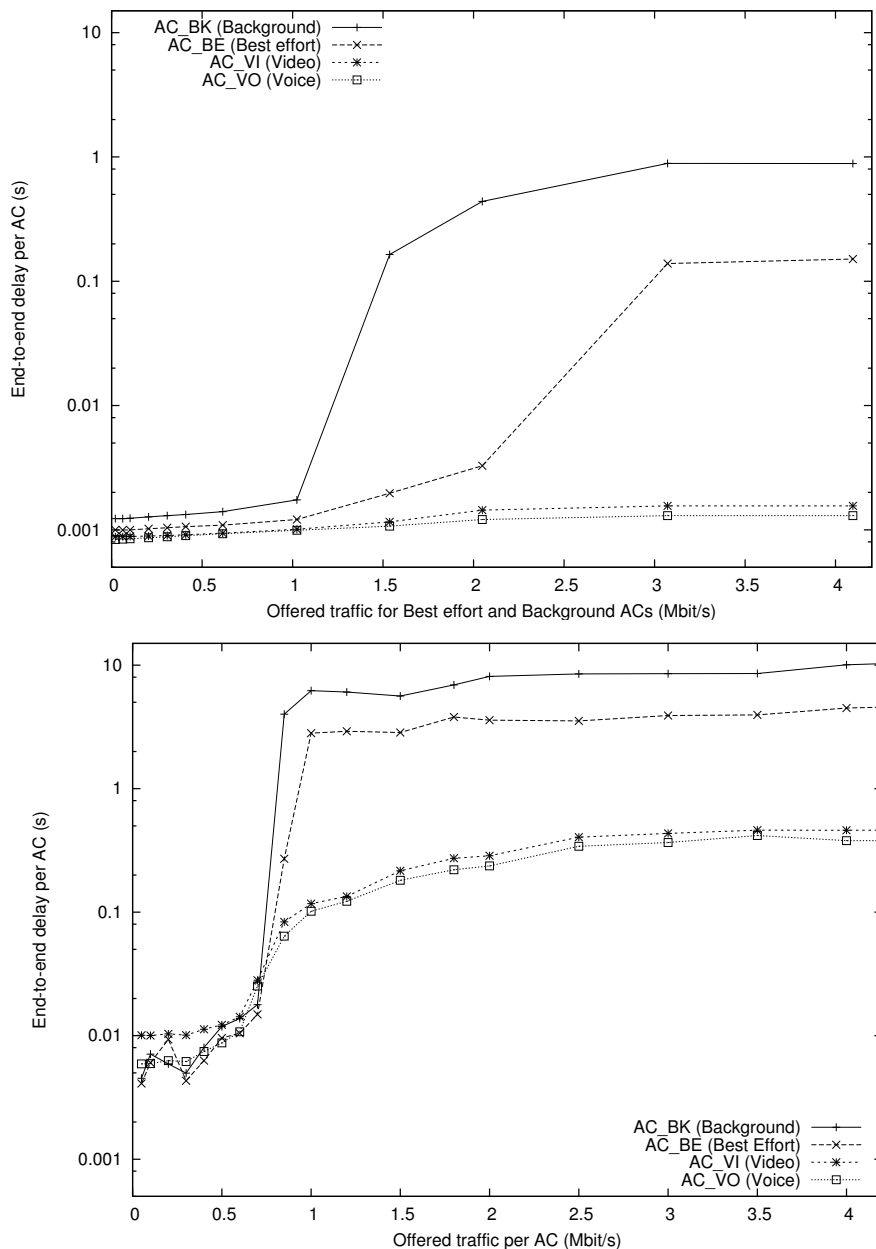


Figure 2.6: End-to-end delay stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (top) and from the testbed (bottom).

solute values, the maximum delay values achieved in the testbed for video traffic (up to 460 ms) can be considered acceptable for real-time videoconferencing applications, though close to the upper limit. Concerning voice sources, the delay values of up to 380 ms are considered prohibitively high for the support of VoIP traffic.

2.6 Conclusions and Future Work

In this paper we assessed the performance of the IEEE 802.11e technology in a multi-hop ad-hoc network testbed, comparing the results obtained with those of the ns-2 simulator.

Our analysis focused on two different issues: (i) assessing the traffic differentiation capabilities of the EDCA channel access mechanism in a multi-hop ad-hoc network, and (ii) determining the degree of QoS robustness experienced by video and voice traffic categories when varying the amount of best effort and background traffic.

With regard to the first issue (i), experimental results showed a significant consistency in terms of overall performance trends. However, the traffic differentiation effectiveness of IEEE 802.11e is partially lost in the testbed when compared to the simulation case in terms of both delay and throughput results.

Concerning the second issue (ii), we found that, contrarily to simulation results, the throughput values for video and voice traffic obtained in the testbed are not maintained stable when increasing traffic from the lower categories (best effort and background), experiencing packet losses of up to 11.5% and 11.8%, respectively. In terms of delay, results showed that the high degree of stability obtained via simulation is no longer achieved; in fact, delay values for QoS traffic experience an increase of about two orders of magnitude compared to simulation results, achieving delay values up to 460 ms and 380 ms for video and voice traffic, respectively.

In general we found that simulation-based results are always optimistic compared to real world performance values, and thus we hope that the work here presented helps at improving IEEE 802.11e simulation models.

Deploying our testbed required solving some technical issues related to wireless card drivers that prevented experimenting with the IEEE 802.11e technology in the ad-hoc mode of operation. We believe this was a main factor hindering the research community from deploying IEEE 802.11e enabled ad-hoc network testbeds; hence, to the best of our knowledge, the research work here presented is pioneer in this research field.

As a future work we intend to implement a distributed admission control system based on probing to offer soft QoS guarantees to real-time traffic.

**Publication B:
Assessing the IEEE 802.11e QoS
effectiveness in multi-hop indoor
scenarios**

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Assessing the IEEE 802.11e QoS effectiveness in multi-hop indoor scenarios*. Ad Hoc Networks, Volume 10, Issue 2. Pages 186 to 198. March 2012.

Abstract

The IEEE 802.11e technology is receiving much interest due to the enhancements offered to wireless local area networks in terms of QoS. Other application fields for this technology are wireless

ad-hoc networks, wireless mesh networks and vehicular ad-hoc networks. In the literature, most of the research works available focusing on the IEEE 802.11e technology offer simulation results alone, being hard to find empirical results of real implementations. Additionally, we consider that the IEEE 802.11e implementation on simulation platforms has not been thoroughly validated using real-life results. In this work we analyze the performance of the IEEE 802.11e technology in real multi-hop ad-hoc networks. With this purpose we first we devise a set of experiments where we compare the results obtained on a small testbed to those from the ns-2 simulation platform. A significant consistency in terms of overall trends is found, although remarkable differences can be appreciated in terms of both delay and throughput results. Afterward we proceed to a full deployment of IEEE 802.11e enabled stations throughout the floor of an university building, and then perform several experiments using both static and dynamic routing. Experimental results show that QoS can be sustained for both Voice and Video traffic in multi-hop ad hoc networks, although dynamic routing protocols can hinder performance by provoking frequent on-off connectivity problems.

3.1 Introduction

Wireless Ad-Hoc Networks are packet radio networks composed by independent and heterogeneous stations that cooperate in routing and packet forwarding tasks, conforming a dynamic multi-hop network.

Interest in these networks has been growing in the last few years since they offer an adequate solution for military and disaster relief scenarios, ubiquitous computing environments, as well as to extend the coverage of networks mainly to provide Internet access

to members of large disperse communities (e.g. university campus, rural areas, etc.)

The proliferation of devices with multimedia and wireless networking capabilities pave the way towards ubiquitous audiovisual communication among peers. Thus, an increasing interest in supporting QoS in ad-hoc networks has arised. To meet this need, the IEEE 802.11e [27] working group has enhanced the IEEE 802.11 standard [26] to provide QoS at the MAC level. MAC level QoS is fundamental in contention-based wireless networks to achieve traffic differentiation in terms of both throughput and delay. In wireless ad-hoc networks this requirement gains further significance since the complex radio interference issues associated with multi-hop channel access cause performance to be poor.

In research works such as [19, 50] authors offer an analysis of IEEE 802.11e in environments where stations communicate with an access point. Other works such as [16] and [23] study the performance of the IEEE 802.11e technology in infrastructureless ad-hoc networks, both static and mobile. The results presented in works such as the ones referred above are usually based on simulation or analytic models. While such an approach is interesting for a first assessment of the performance and viability of QoS solutions, they tend to be optimistic and miss several problems associated with real-life implementations, such as hardware limitations, background noise and interferences, software restrictions and, last but not the least, coding bugs in the wireless cards' driver.

In this paper we setup an IEEE 802.11e enabled testbed using native Linux drivers that allows us to assess the performance of this technology in multi-hop ad-hoc network environments. We first compare the values obtained in the testbed against simulation results obtained with the ns-2 simulator. Experiments evidence some significant performance differences in terms of throughput and delay. Afterward we deploy several terminals throughout the

floor of a building and proceed to evaluate QoS effectiveness in a realistic test case.

The structure of this paper is the following: in the next section we refer to some related works regarding IEEE 802.11e testbeds. In section 3.3 we briefly introduce the IEEE 802.11 technology, as well as the enhancements proposed in the current IEEE 802.11e standard to support QoS. In section 3.4 we describe the characteristics of our testbed in terms of both hardware and software configurations. A result comparison of testbed experiments vs. ns-2 simulation is then presented in section 3.5. Section 3.6 presents a performance evaluation of IEEE 802.11e in a distributed indoor environment characterized by multiple hops and significant physical barriers using both static and dynamic routing. Finally, section 3.7 concludes the paper.

3.2 Related Works

In the literature we can find a few recent works where the authors setup a testbed to evaluate the performance of the IEEE 802.11e technology. In [38] Ng et al. setup a testbed to demonstrate that the 802.11e parameters CWmin, TXOP and AIFS work largely in line with analytic and simulation predictions. Additionally, they use the testbed to demonstrate some known problems with TCP's performance caused by cross-layer interaction between the TCP congestion control algorithm and the MAC layer CSMA/CA contention mechanism, as well a possible solution to that problem.

Dangereld et al. [21] have studied the delay in the context of protecting a voice call competing against data traffic. In particular, they present experimental measurements which demonstrate that the use of 802.11e offers a practical solution to successfully deliver

quality of service to voice traffic in a mixed voice/data environment.

Banchs et al. [13] focus on applications of EDCA, i.e., the *enhanced distributed channel access* proposed by the IEEE 802.11e standard. In particular, the target applications are traffic engineering and service guarantees. Experimental results show that EDCA can effectively be used for traffic engineering purposes. However, the goal of providing service guarantees with EDCA is shown to be more challenging.

Vasilos and Stamatakis [46] investigate the optimal selection of minimum contention window values to achieve proportional fairness in a multirate IEEE 802.11e test-bed. The test-bed experiments show that when stations have different transmission rates and the same weight, proportional fairness achieves higher performance than the time-based fairness approaches, in terms of both aggregate utility and throughput.

All of the aforementioned testbeds were configured for the infrastructure mode alone, using Atheros wireless cards (either PCI or PCMCIA) and the MadWifi driver [4] for testing. Notice that this driver does not fully support the IBSS mode of operation, which has hindered researchers from testing the benefits of the IEEE 802.11e technology in ad-hoc network environments. Additionally, the driver does not support USB devices nor most of the wireless cards integrated in laptops.

To the best of our knowledge, our work is pioneer in terms of performance assessment of the IEEE 802.11e technology in a multi-hop ad-hoc network environment. Moreover, we rely on enhanced versions of standard wireless drivers for the Linux kernel that support state-of-the-art USB and integrated wireless cards, which favors a widespread adoption of this technology among both researchers and regular users.

3.3 Overview of the IEEE 802.11 Standard

The IEEE 802.11 standard [26] is a technology whose purpose is to provide wireless access to local area networks. Stations using this technology access the wireless medium using either the Point Coordination Function (PCF) or the Distributed Coordination Function (DCF).

The Point Coordination Function is a centralized access mode optionally used in a Basic Service Set (BSS) when a point coordinator (PC) is available. The PC is typically an *access point* (AP), and so the stations are said to operate in infrastructure mode. When relying on the PCF, contention-free periods (CFP) and contention periods (CP) alternate over time. The regular generation of beacons allows stations to associate and synchronize with the PC. Typically a CFP is started after a beacon management frame, followed by a CP; together they conform a superframe. During the CFP there is no contention, meaning that stations are merely polled by the PC. The CF-End control frame is transmitted by the PC to indicate the end of the CF period, and the beginning of the CP. During the CP stations access the medium using the DCF.

The Distributed Coordination Function (DCF) uses a listen-before-talk scheme named *carrier sense multiple access* (CSMA) with *collision avoidance* (CA). It is used by stations in a BSS during the CP and also by stations in an IBSS operating in ad-hoc mode. The CSMA technology distributes the medium access task among all stations, making every station responsible for assuring the delivery of *MAC service data units* and reacting to collisions. The collision avoidance (CA) scheme is used to reduce the probability of collisions between different stations. To achieve this it applies a backoff procedure before initiating a transmission if the medium was not

found to be previously idle. Stations select a random number of slots to wait before transmission in the interval between 0 and the current *contention window* (CW) value. The value for CW is set initially to the minimum value for the radio technology being used (CW_{min}), being increased when consecutive collisions occur up to a maximum value (CW_{max}).

The CSMA/CA mechanism shows good adaptation to different numbers of transmitters, and probabilistically shares the channel equally among all competing stations. However, it offers no mechanisms to perform traffic differentiation, making QoS support practically unfeasible. The IEEE 802.11e working group was created to add QoS support to the original IEEE 802.11 standard, and in 2005 a new international standard [27] was released.

3.3.1 IEEE 802.11e: MAC enhancements for QoS

The IEEE 802.11e working group has extended the IEEE 802.11 MAC in order to provide QoS support. This standard introduces the *hybrid coordination function* (HCF) which defines two new medium access mechanisms to replace PCF and DCF. These are the *HCF controlled channel access* (HCCA) and the *enhanced distributed channel access* (EDCA).

With the HCF there may still exist a contention period and a contention-free period in a superframe, but now the HCCA is used in both periods, while the EDCA is used only during the CP. This new characteristic of HCF obviates the need for a CFP since it no longer depends on it to provide QoS guarantees.

With 802.11e, the *point coordinator* is replaced by a *hybrid coordinator* (HC) which also resides in an AP. A BSS including a HC is referred to as a QBSS. In this paper we focus on ad-hoc networks and, therefore, we are only interested in 802.11e stations imple-

menting EDCA. For more information on HCs, the HCF and the HCCA refer to [27].

Concerning 802.11e enabled stations forming an ad-hoc network, these must implement the EDCA algorithm. The 802.11e QoS support is achieved through the introduction of different *access categories* (ACs), and their associated backoff entities.

Contrarily to the legacy IEEE 802.11 stations, where all packets have the same priority and are assigned to a single backoff entity, IEEE 802.11e stations have four backoff entities (one for each AC) so that packets are sorted according to their priority. The different access categories available in IEEE 802.11e stations are: Voice (AC_VO), Video (AC_VI), Best-effort (AC_BE) and Background (AC_BK).

Each backoff entity has an independent packet queue assigned to it, as well as a different parameter set. In IEEE 802.11 legacy stations, this parameter set was fixed, and so the inter-frame space was set to DIFS and the CWmin and CWmax were set to 15 and 1023 respectively (for IEEE 802.11a/g). With IEEE 802.11e the inter-frame space is arbitrary and depends on the access category itself (AIFS[AC]). We also have AC-dependent minimum and maximum values for the contention window (CWmin[AC] and CWmax[AC]). Additionally, IEEE 802.11e introduces an important new feature referred to as transmission opportunity (TXOP). A TXOP is defined by a start time and a duration; during this time interval a station can deliver multiple frames consecutively without contention with other stations. This mechanism, also known as *contention-free bursting* (CFB), increases global throughput through a higher channel occupation. An EDCA-TXOP (in contrast to an HCCA-TXOP) is limited by the value of TXOPLimit, which is a parameter defined for the entire QBSS and that also depends on the AC (TXOPLimit[AC]).

Table 3.1 presents the default MAC parameter values for the dif-

ferent ACs. Notice that smaller values for the AIFSN, CW_{min} and CW_{max} parameters result in a higher priority when accessing the channel; relative to the TXOPLimit, higher values result in larger shares of capacity and, therefore, higher priority.

Table 3.1: IEEE 802.11e MAC parameter values for a IEEE 802.11a/g radio

<i>Access category</i>	<i>AIFSN</i>	<i>CW_{min}</i>	<i>CW_{max}</i>	<i>TXOPLimit (ms)</i>
AC_BK	7	15	1023	0
AC_BE	3	15	1023	0
AC_VI	2	7	15	3.008
AC_VO	2	3	7	1.504

The relation between AIFS[AC] and AIFSN[AC], is the following:

$$AIFS[AC] = SIFS + AIFSN[AC] \times aSlotTime \quad (3.1)$$

where *SIFS* is the shortest inter-frame space possible and *aSlotTime* is the duration of a slot. As defined by the standard, the *AIFSN[AC]* parameter must never be less than 2 to avoid interference with normal AP operation.

For applications to take advantage of the IEEE 802.11e technology, datagrams should have their IP TOS header field set according to the desired user priority. When delivered to an IEEE 802.11e enabled wireless card driver, those datagrams will be handled according to the priority defined. The mapping between user priorities (network layer) and the four medium access categories defined by the IEEE 802.11e standard (MAC layer) is shown in Table 3.2 for reference.

Table 3.2: User Priority to IEEE 802.11e Access Category Mapping (according to IEEE 802.1D)

User Priority	Designation	Access Category
1	BK (Background)	AC_BK
2	BK (Background)	AC_BK
0	BE (Best-effort)	AC_BE
3	EE (Video/Excellent-effort)	AC_BE
4	CL (Video/Controlled Load)	AC_VI
5	VI (Video)	AC_VI
6	VO (Voice)	AC_VO
7	NC (Network Control)	AC_VO

3.4 Testbed Setup

In this section we describe the hardware used and the software configuration details for the real testbed being deployed. We also describe in detail how throughput and delay measurements are performed. The testbed setup information provided in this section applies to the experimental results presented in sections 3.5 and 3.6.

3.4.1 Hardware and software details

To create a IEEE 802.11e enabled multi-hop ad hoc network we relied on 11 computers, where 8 of them were low-cost Asus EeePC 901 netbooks, 1 was a high performance Lenovo laptop, and the remaining 2 were fully equipped desktop PCs.

In terms of wireless cards, we opted for using IEEE 802.11e enabled cards using the Ralink chipset [7]. Our decision was based

on the availability of the source code for the Ralink wireless card drivers for Linux, a basic requirement to allow performing the required enhancements. In particular, on the Asus netbooks we used their integrated wireless card (Ralink RT 2860 chipset), and on the remaining terminals we used either D-Link DWA-140 or Linksys WUSB600N-EU wireless USB cards, both based on the Ralink RT 2870 chipset.

All the terminals involved in the testbed had a GNU/Linux operating system installed, either version 2.6.27 (netbooks and laptop) or 2.6.24 (desktop). The Ralink wireless card drivers used were 1.7.0.0 for the RT2860 chipset, and 1.3.1.0 for the RT2870 chipset¹.

The main problem encountered when attempting to deploy an infrastructureless wireless network using the aforementioned hardware was the lack of support for IEEE 802.11e in the ad-hoc mode of operation. Such setback evidences the lack of interest shown by hardware manufacturers in this issue, requiring significant enhancements to the original drivers in order to make EDCA channel access operative in the infrastructureless mode. Additionally, further enhancements were made to add support for the 2.6.27 Linux kernel, to allow fixing the data rate at 54 Mbit/s (a basic requirement for the experiments of section 3.5 to make results comparable with those of ns-2), as well as to correct some detected bugs.²

3.4.2 Throughput and delay measurements

While simulation traces allow to easily obtain all required metrics, in a real testbed the process is more complex. In particular, measuring the throughput required enabling the *tcpdump* capture

¹These drivers are available at:
<http://www.ralinktech.com/ralink/Home/Support/Linux.html>

²Drivers are available upon request.

tool, based on *libpcap* [53], on all traffic destinations (stations D_i). The captured traffic was then filtered to discriminate throughput among the different QoS categories; finally, the values provided by the different destinations were aggregated to obtain per-category throughput values.

Concerning delay, the measurement process is significantly more complex since sender/receiver clocks are not synchronized. Different delay measurement possibilities were discussed in [21], and for this testbed we adopted the solution proposed in [18], where a Fast Ethernet backbone is used to forward traffic back to the source, thus allowing to compute delay using a same clock. To make these measurements straightforward we developed our own UDP traffic generator. This traffic generator includes a time-stamp at the beginning of the packet's payload and, when receiving it again through the feedback channel, clock differences allow to compute delay.

To make sure that the Fast Ethernet return path did not affect delay measurements in a significant manner we calculated the additional delay introduced by the wired channel for different load values. The experimental results are shown in Figure 3.1.

The maximum load measured on the return channel during our testbed experiments never surpassed 5 Mbit/s, which means that the additional delay introduced by the Fast-Ethernet connection is rather small. Nevertheless, we used these results to obtain unbiased estimations of all delay values, thus measuring the delay associated with the wireless counterpart much more precisely.

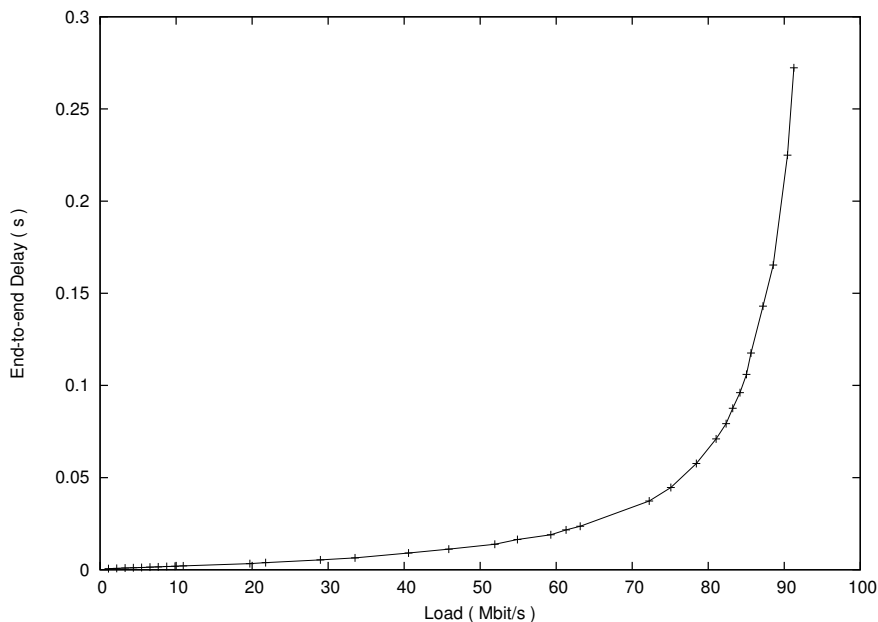


Figure 3.1: Load vs. delay values for the Fast-Ethernet link used as return path.

3.5 Comparative analysis of simulation and testbed results

In this section we present a performance assessment study of the IEEE 802.11e technology in a multi-hop ad-hoc network environment where we compare simulation results against testbed results.

We have split our analysis into two sets of experiments. In the first set of experiments (section 3.5.2) our purpose is to assess the traffic differentiation capabilities of the EDCA channel access mechanism in the proposed multi-hop ad-hoc network. In the second set of experiments (section 3.5.3) our goal is to determine the degree

of robustness and stability that the two highest access categories (Video and Voice) experience when varying the amount of best effort and background traffic.

3.5.1 Scenario definition

Figure 3.2 shows the multi-hop ad-hoc network scenario we have selected for our comparative study. The scenario includes four source-destination pairs, as well as three intermediate stations, thus conforming a total of 11 stations. Unless stated otherwise, all traffic sources are set to generate the same data rate in all four ACs, and the traffic type chosen is constant bitrate (CBR) with packet size fixed at 512 bytes to fully control the load being injected. In our experiments we avoid synchronization among traffic sources to prevent results from being biased towards a worst case scenario; this was achieved by adding a small jitter to the inter-packet interval.

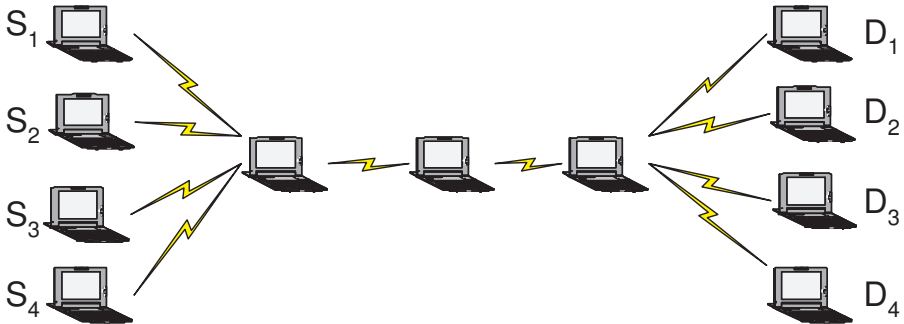


Figure 3.2: Multi-hop ad-hoc network scenario used for testing.

The simulation results used for comparison were obtained using the ns-2 simulator [30]. It was configured with the same scenario, and all nodes have an IEEE 802.11g radio operating at a fixed data

rate of 54 Mbit/s. The distance between stations that are one hop away is of 200 meters, and simulation takes place within an area sized 1000×400 meters. Notice that the carrier sensing range is of 550 meters, meaning that interferences make spatial reuse to be significantly reduced (in fact, it is null for the center station).

For our testbed we could not use exactly the same settings due to space and radio range restrictions. In fact, to operate efficiently with IEEE 802.11g cards at a fixed data rate of 54 Mbit/s, stations must be at most a few meters away. Thus, in our testbed all the stations are within the same room (separated by about 1 meter), and the topology shown in Figure 3.2 was enforced logically through an adequate configuration of the firewall using the *iptables* tool [9]. Despite the total channel capacity of the testbed is reduced with respect to the simulator, we consider that it offers the best trade-off when assuming basic feasibility requirements. Nevertheless, when performing result comparisons, we shall take this restriction into consideration.

Due to the presence of other sources of noise in the 2.4 GHz band, we used the Wi-Spy spectrum analyzer [5] to aid us in determining the best channel for experimentation. The characteristics of the channel used are shown in Table 3.3 for the sake of completeness. Those values were obtained with all the stations participating in the testbed turned off.

Table 3.3: Characteristics of the wireless channel selected (obtained with Wi-Spy 2.4GHz spectrum analyzer).

<i>Parameter</i>	<i>Value range</i>
Average Floor (dBm)	-85.94
Average Peak (dBm)	-79.13
Duty Cycle	11.06

We ran the same set of tests using both the wireless testbed de-

scribed and the ns-2 simulator. All experiments run for 120 seconds, being that the first and the last 10 seconds of each experiment were discarded. The chosen performance metrics are the throughput and the end-to-end delay values. Each experiment was repeated 20 times, which allowed us to determine, with a degree of confidence of 95%, that the maximum error for the mean is under 5% for all the values presented in this section.

With respect to the IEEE 802.11e technology, the stations used in our testbed adopt the EDCA channel access parameters shown in Table 3.1, and the contention-free bursting (CFB) mechanism is turned off. The same applies to virtual stations participating in the ns-2 based simulation experiments. Notice that ns-2 does not currently include support for IEEE 802.11e, which required installing the extensions developed by Wietholter and Hoene [48].

3.5.2 Traffic differentiation analysis

When a network is very lightly loaded QoS mechanisms are both ineffective and unnecessary. However, as we increase traffic to moderate/high loads, QoS mechanisms become fundamental to discriminate among the different types of traffic. Therefore, in the current set of experiments we gradually increase the traffic load by the same amount for all four MAC categories defined by the IEEE 802.11e standard.

Figure 3.3 shows the effectiveness of the IEEE 802.11e technology in terms of traffic differentiation at different loads. Comparing the results obtained with the simulator with those obtained in our testbed allows us to draw several conclusions: (i) in a real testbed the throughput values achieved before reaching the saturation points do not always present a matching with the offered load due to more frequent losses; this is especially noticeable for

3.5 Comparative analysis of simulation and testbed results

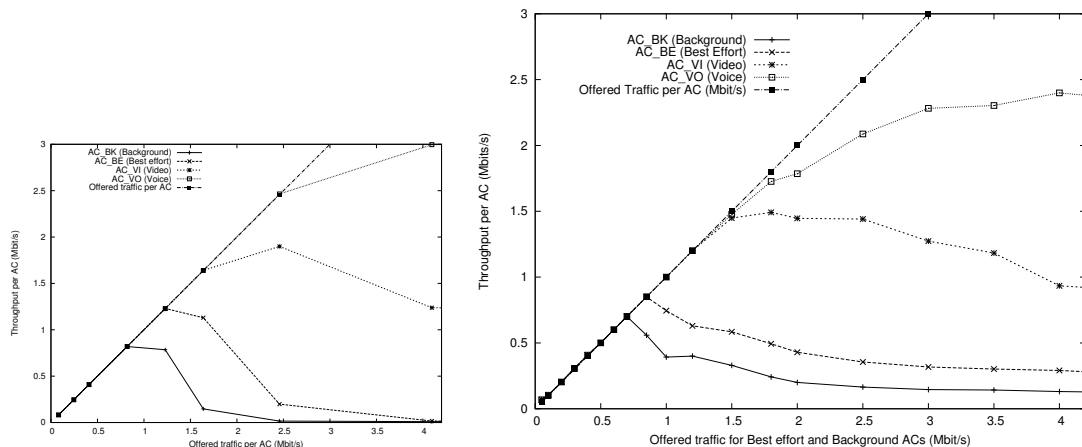


Figure 3.3: Throughput values achieved via simulation (left) and in our testbed (right).

the voice and video traffic categories, and is typically associated with channel noise and external sources of interference; (ii) in the testbed the throughput for all traffic types decays more abruptly when the saturation point for each access category is reached; additionally, we also observe that the particular saturation point for each access category is reduced compared to simulation results. This effect is related to the lower channel capacity of the testbed environment due to the overlapping in terms of radio coverage and the presence of beacons; (iii) the traffic differentiation effectiveness of IEEE 802.11e is partially lost in the testbed compared to the simulation case, which allows to avoid starvation of best effort and background traffic by reducing QoS traffic (mostly in the video access category).

Figure 3.4 shows the results for this same set of experiments, but now focusing on the end-to-end delay metric instead. As expected, the most significant performance issues referred above guard a

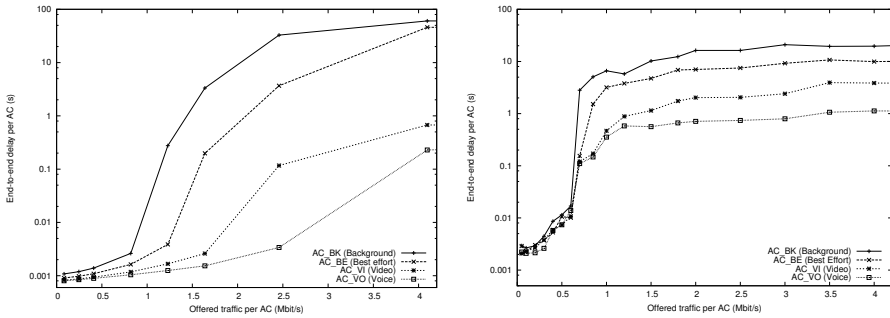


Figure 3.4: End-to-end delay values achieved via simulation (left) and in our testbed (right).

strict relationship with the end-to-end performance. In particular, we should remark that: (i) contrarily to the simulation experiments, the delay values obtained in the testbed experience a much more steep growth for low load values (notice that a logarithmic scale is used for the vertical axis); and (ii) when reaching moderate/high saturation levels the differentiation capabilities of IEEE 802.11e in the testbed experience a significant loss of effectiveness compared to simulation results. In particular, average delay values for video and voice traffic are increased by one order of magnitude; such difference should be considered quite significant and taken into consideration when designing QoS architectures based on the IEEE 802.11e technology.

3.5.3 QoS robustness analysis

We now present the results related to the second set of experiments, where the aim is to assess whether the IEEE 802.11e technology is able to offer steady performance levels to active QoS connections when increasing non-QoS traffic. With this purpose we fix the aggregated load for the video and voice traffic at 1 and 0.5

3.5 Comparative analysis of simulation and testbed results

Mbit/s, respectively, increasing load levels for the best-effort and background traffic categories alone.

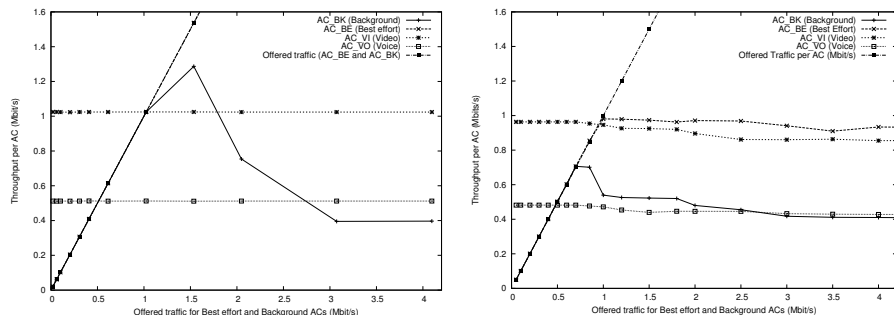


Figure 3.5: Throughput stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).

Figure 3.5 shows the throughput results obtained using the wireless testbed and the ns-2 simulator. It is noticeable that in the testbed the throughput values for video and voice traffic are not maintained stable, experiencing packet losses of up to 11.5% and 11.8%, respectively. This is in clear contrast with simulation results, where throughput levels are maintained practically unaltered for all load values. As a secondary issue, we noticed that the throughput values for best effort traffic under saturation differ greatly, leveling out at 2.4 Mbit/s in the ns-2 simulator (not visible due to scale), while in the testbed they level-out at about 0.9 Mbit/s. The observed differences, are mainly due to the lower overall channel capacity of the testbed environment, as explained earlier.

In terms of end-to-end delay, the results presented in Figure 3.6 guard some resemblance with the ones of Figure 3.4, and again evidence the difficulties at maintaining reduced delay values for video and voice traffic in a real testbed. The most noticeable difference is detected in the testbed results at load values close to 0.7

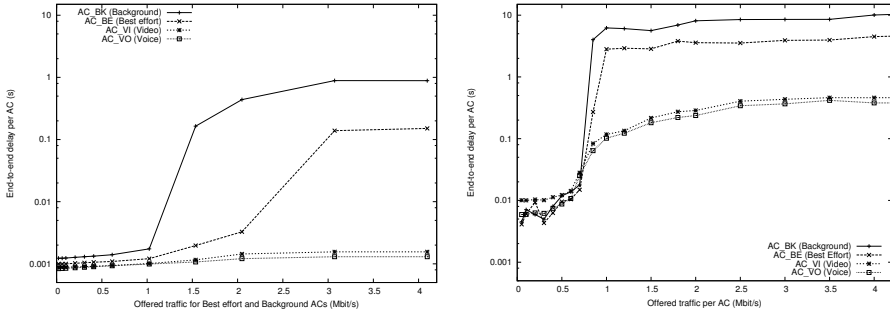


Figure 3.6: End-to-end delay stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).

Mbit/s, where a sharp increase in the delay values of all traffic categories occurs. In particular, video and voice traffic experience an increase of about two orders of magnitude compared with simulation results, while the increase for both best effort and background traffic is limited to one order of magnitude. Thus, the high degree of delay stability obtained via simulation is lost. In terms of absolute values, the maximum delay values achieved in the testbed for video traffic (up to 460 ms) can be considered acceptable for real-time videoconferencing applications, though close to the upper limit. Concerning voice sources, the delay values of up to 380 ms are considered prohibitively high for the support of VoIP traffic.

3.6 Evaluating IEEE 802.11e in a realistic indoor scenario

The results presented in the previous section were made, for comparison purposes, with all terminals concentrated in a same room. This allowed us to use IEEE 802.11g and fix the data rate at 54

Mbit/s as intended, thus achieving the highest possible degree of similitude with the simulation settings.

In this section we drop the constraints set for the previous section, allowing to take full advantage of the wireless cards by (i) enabling IEEE 802.11n mode and using a bandwidth of 40 MHz, although taking frequencies from the 2.4 GHz band alone, and (ii) enabling the auto-rate adjustment mode, used by default in 802.11, whereby the network interface will automatically pick the best data rate possible according to Signal-to-Noise Ratio (SNR) conditions.

Our research purposes were twofold: in the one hand we wanted to assess the performance differences experienced in a realistic scenario where greater distances between nodes, along with physical barriers, caused the SNR to be quite inferior to the overlapped coverage scenario tested before. On the other hand, we wanted to compare static against dynamic routing to evaluate how route selection and route variability affected QoS performance.

3.6.1 Node deployment and configuration issues

For our tests we deployed eight Asus EEEPC laptops throughout a same building floor, enforcing multi-hop connectivity in a natural manner due to distance and the presence of brick walls, wooden doors, double glass windows and metal filing cabinets in each room.

Figure 3.7 shows a detailed map of the third (and last) floor of our School of Informatics at Valencia, Spain. Dark dots tagged with letters represent mobile terminals, and the thick lines connecting them represent the wireless links; only good links, that is, links which are sustained in time, are represented. As expected, physical obstacles become a serious hindrance to signal propagation in the 2.4 GHz band; in particular, it is curious to see how link C-G,

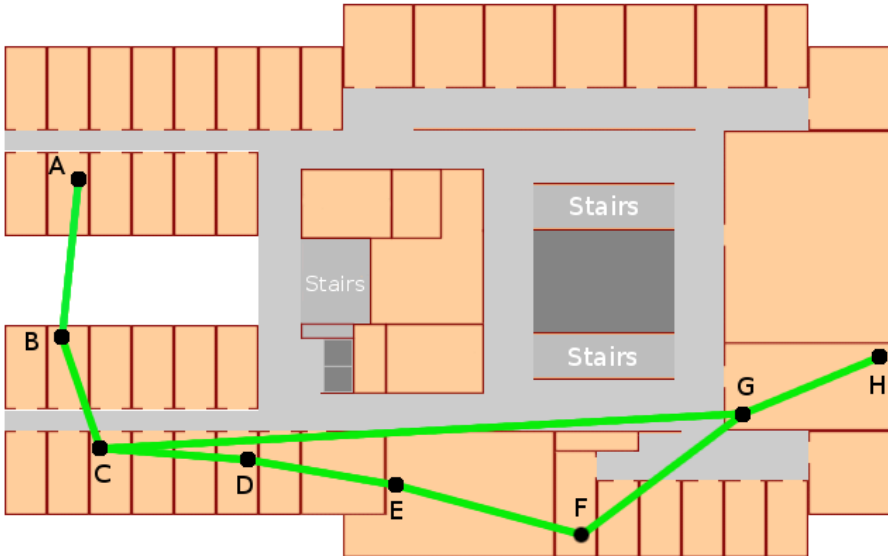


Figure 3.7: Map of the building floor where tests were made and node deployment details. Approximate floor dimensions (maximum): 60×40 meters.

which mainly traverses open space through a corridor, is much longer than any other link; in this case it is also the weakest.

For the experiments that follow we adopted a strategy similar to the previous section, whereby a wired network is used for station management and control purposes. The wired network is Fast-Ethernet based, and connects stations with a control application running on a remote PC through SSH.

With respect to traffic generation, in our tests we will evaluate the QoS conditions experienced by a bidirectional videoconferencing session where synthetic video and voice traffic are generated as independent UDP flows so as to take advantage of the Video and Voice MAC access categories of IEEE 802.11e. The video data rate

is of 1 Mbit/s in each direction, while voice data rate is of 15 kbit/s in each direction.

In addition to QoS traffic, we also start an FTP/TCP flow between both endpoint involved in the videoconference as best effort traffic. The purpose of background traffic is twofold: (i) to increase the degree of congestion to match a more realistic situation, and (ii) to take advantage of TCP's bandwidth greediness to have an estimate of the residual bandwidth available for best effort traffic on the end-to-end connection.

When static routing is used we fill-in the routing tables of all nodes by adopting a shortest path criteria according to the topology shown in Figure 3.7. Dynamic routing tests rely on the well-known OLSR protocol [49] instead.

All the results presented in the sections that follow are average values where we take the results from 10 experiments and discard the two best and the two worst ones, retrieving the mean for the remaining six. Each of our experiments lasts for about 300 seconds.

3.6.2 Performance results

We now present the performance results obtained in our indoor test scenario. For UDP traffic (video and voice) we obtain the packet delivery ratio, while for TCP traffic we obtain the throughput.

Figure 3.8 shows the results achieved at one hop. We can see that C-G is the weakest link due to the greater distance between stations. On the contrary, G-H and B-C are the links achieving better performance. This can be seen from both charts by noticing that the UDP packet delivery ratio is near optimum, while TCP traffic also achieves an important share of bandwidth (up to 37 Mbit/s for the BC link).

By comparing static against dynamic routing we can appreciate significant differences for links A-B, C-G and D-E, especially in terms of video traffic. Since the delivery ratio for voice is sustained, the losses experienced can not be related to route failure, being instead related to channel contention caused by the increase of routing-related control traffic.

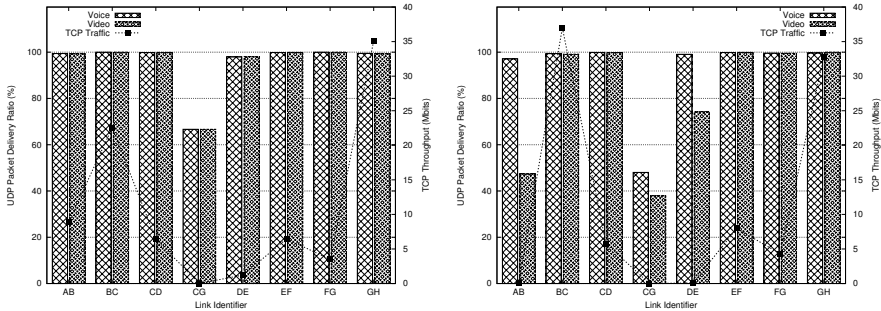


Figure 3.8: UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at one hop using static (left) and dynamic (right) routing.

The results for the two-hop experiments are shown in Figure 3.9. For the static routing scenario we observe that the connections associated with endpoints BG, CF, CH, and DG experience a worse performance than the rest. This occurs because all of them make use of the C-G link to reach the destination, which was shown to offer the worst performance in the previous section. The remaining connections experience a reasonably good performance, although increased channel contention now provokes more losses. The most representative situation is observed for the AC connection, where QoS traffic suffers from losses between 1% and 2%, although the amount of bandwidth available is enough to achieve a throughput of about 5.3 Mbit/s.

When analyzing the dynamic routing scenarios we can observe

3.6 Evaluating IEEE 802.11e in a realistic indoor scenario

that, again, performance is degraded compared to the static routing case. We must take into consideration that now OLSR will determine the route dynamically throughout time, being possible that the route used differs from the static routing test. By analyzing Figure 3.9 two situations are detected: in most cases (connections AC, BD, BG, CE, CF, CH, DG, and FH) we observe a pattern similar to the static routing scenario, with the differences that performance is slightly inferior for voice traffic, and significantly inferior for video traffic. However, there are a few other cases (connections DF and EG) where the routing protocol used (OSLR) is not able to sustain the route during the entire test period, thus causing the performance to be much worse than for the static routing case.

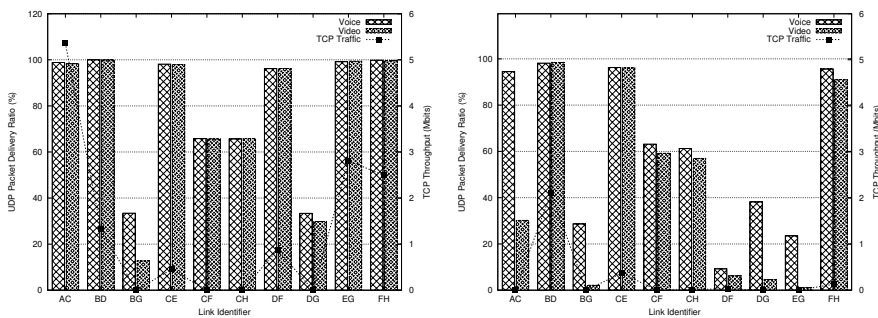


Figure 3.9: UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at two hops using static (left) and dynamic (right) routing.

Similar degradation trends were found for the three and four hops cases (see Figures 3.10 and 3.11). It also is interesting to see that connections avoiding the weakest link (C-G) are able to maintain video and voice loss levels at a minimum, with a packet arrival ratio near 100%. To confirm this just check connections AD, BE and EH for the three-hops case, as well as connection AE for the four-hops

case. Such results clearly show that the IEEE 802.11e technology is able to sustain QoS at high levels, even when operating in a multi-hop fashion, as long as the none of the links is weak. Otherwise performance will drop down to unacceptable levels, with QoS traffic experiencing significant losses.

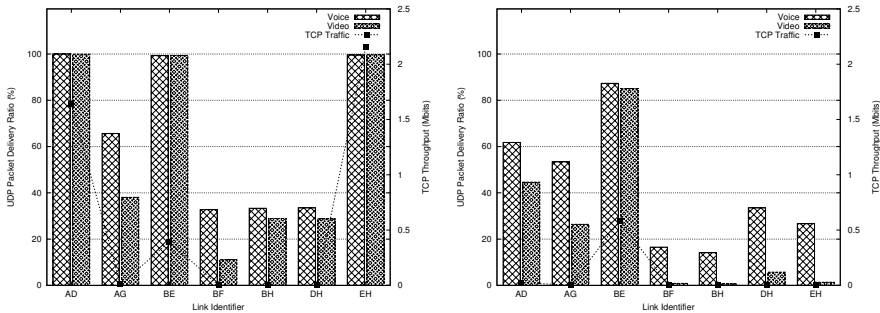


Figure 3.10: UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at three hops using static (left) and dynamic (right) routing.

Focusing on the differences between static and dynamic routing for the three and four hops cases, we find that, as the number of hops grows, route instability will tend to increase. In particular, the results we obtained show that the packet delivery ratio is always below 90% when transmitting at a three hop distance, and it is below 60% when transmitting at a four hop distance.

TCP-based best effort traffic is also greatly affected by route instability, suffering from starvation (throughput of 12 kbit/s in the best case) when transmitting at a four-hop distance.

All the results presented above clearly show that routing protocols for MANETs still require important improvements in order to achieve the stability and efficiency required to operate in a multi-hop ad hoc network. In particular, OSLR presents route misbehavior in indoor scenarios even in the absence of mobility.

3.7 Conclusions and Future Work

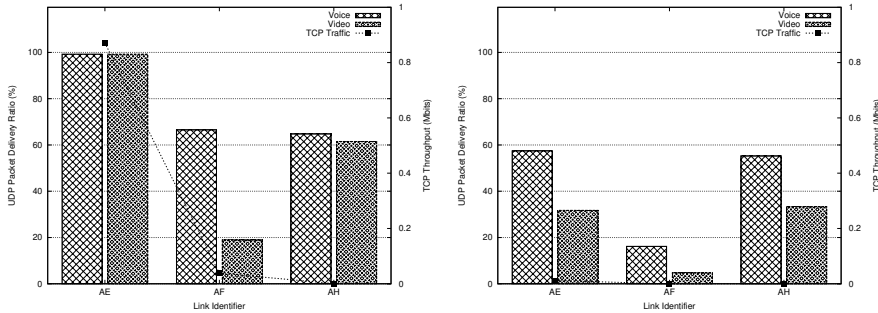


Figure 3.11: UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at four hops using static (left) and dynamic (right) routing.

3.7 Conclusions and Future Work

In this paper we assessed the performance of the IEEE 802.11e technology in a multi-hop ad-hoc network testbed.

We first performed a comparison of testbed results against the simulation outcome as provided by the ns-2 simulator in a specific setting. Our analysis focused on two different issues: (i) assessing the traffic differentiation capabilities of the EDCA channel access mechanism in a multi-hop ad-hoc network, and (ii) determining the degree of QoS robustness experienced by video and voice traffic categories when varying the amount of best effort and background traffic.

With regard to the first issue, experimental results showed a significant consistency in terms of overall performance trends. However, the traffic differentiation effectiveness of IEEE 802.11e is partially lost in the testbed when compared to the simulation case in terms of both delay and throughput results. Concerning the second issue, we found that, contrarily to simulation results, the throughput values for video and voice traffic obtained in the testbed are not

maintained stable when increasing traffic from the lower categories (best effort and background), experiencing packet losses of up to 11.5% and 11.8%, respectively. In terms of delay, results showed that the high degree of stability obtained via simulation is no longer achieved; in fact, delay values for QoS traffic experience an increase of about two orders of magnitude compared to simulation results, achieving delay values up to 460 ms and 380 ms for video and voice traffic, respectively. In general we found that simulation-based results are always optimistic compared to real world performance values.

In the second part of our study we deployed an indoor ad hoc network where terminals were distributed throughout the floor of an university building sized 60×40 meters, and with a significant amount of physical obstacles. For this scenario we relied on IEEE 802.11n radio communications for improved performance. Experimental results for a bidirectional videoconference between different pairs of stations have showed that the IEEE 802.11e is able to retain its effectiveness despite the multi-hop constraint for communication up to four hops as long as the quality of all links is reasonably good. Our analysis also showed that, compared to static routing, OLSR-based dynamic routing is prone to cause intermittent connectivity problems that hinder QoS, despite no mobility is introduced. Such routing problems have in impact both on QoS traffic, which experiences significant losses, and on TCP-based best effort traffic, which suffers from starvation.

Overall, we consider that to successfully achieve high levels of QoS support when deploying real multi-hop ad hoc networks based on IEEE 802.11e there are still several technical challenges that must be solved. The most critical are a stable and QoS-aware routing protocol implementation, distributed admission control systems, and 802.11 interface drivers that are optimized for the ad hoc mode of operation. Hence, in future works we will address these research

3.7 Conclusions and Future Work

challenges.

**Publication C:
Distributed Admission Control in
802.11e-Based MANETs: from
Theory to Practice**

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Distributed Admission Control in 802.11e-Based MANETs: from Theory to Practice*. IFIP Wireless Days (WD), 2011. Pages 10 to 12. October 2011

Abstract

Providing Quality of Service (QoS) in wireless ad-hoc networks is an extremely complex task due to issues such as mobility, channel contention, and interference. Over the past years, several algorithms have been presented with the aim of providing QoS support in these environments; specifically, a lot of work has been done in

the creation of a reliable admission control mechanism since it has been proved to be a fundamental element for QoS support in ad-hoc networks. In this paper we present a real implementation of DACME, a distributed admission control system for mobile ad-hoc networks, and we test its effectiveness in an IEEE 802.11e enabled testbed. Experimental results show that the solution developed is able to achieve good QoS levels, offering sustained bandwidth and bounded delay. Also, since DACME operates on an end-to-end basis, no restrictions are imposed on intermediate nodes.

4.1 Introduction

Nowadays, mobile ad-hoc networks (MANETs) provide a cheap and infrastructureless form of communication. When combined with an appropriate routing protocol, the IEEE 802.11 standard allows to easily deploy a MANET, which can be very useful in areas where the provision of a central infrastructure is limited or lacking. At the application level, typical MANET users share messages and collaborate with each other [17].

More recently, interest has grown in enabling MANETs to support real-time services such as videoconferencing or VoIP calls among participants. However, this type of network lacks a good quality of service (QoS) support. In this scope, the IEEE 802.11e working group [27] provided QoS support at the MAC level, although targeting mostly wireless LANs. This extension to the original IEEE 802.11 standard introduces four new traffic categories: Voice, Video, Best Effort, and Background (ordered according to their priority).

Despite the enhancements that IEEE 802.11e has brought, it is still not enough when facing QoS flow concurrency. In fact, one

of the most crucial components of a system attempting to provide QoS guarantees is the Admission Control Module.

In the literature we can find several admission control (AC) algorithms for MANET environments. Following the guidelines provided by the survey made by Hanzo et al. [28], the different AC algorithms available can be divided in two large groups: routing coupled and routing decoupled. In the first group AC algorithms require modifying the routing algorithm to support the AC extension. This strategy has some benefits, such as shorter admission times and less overhead of the AC protocol. However, this group also presents several drawbacks, such as being coupled with a specific routing algorithm and imposing strong requirements on nodes, forcing every node in the network to adopt the modified routing protocol, to support the AC module.

With respect to routing decoupled algorithms, their main advantage is that they allow using any routing protocol for MANET environments. Within the routing decoupled algorithms group, an additional division can be made between stateful and stateless protocols. Stateful algorithms save certain information about the state of the links in every node. Such strategy allows AC algorithms such as INSIGNIA [32] to store information about QoS conditions relative to past flows, and decide, based on this information, whether to accept or reject the flow. Thus, similarly to what occurs for routing coupled protocols, they impose several restrictions on intermediate nodes. Within the stateless, routing decoupled protocols group, we can find solutions such as DACME [15], which do not impose any restriction on intermediate nodes since there is no need to store information about past flows, nor does require nodes to have a high computing power.

The main drawback associated with all the aforementioned solutions is that they have been tested on simulated environments, and, to the best of our knowledge, none has been implemented and

tested in a real environment. In this paper we will develop and test DACME. We chose this proposal since it has been proved to be a powerful and efficient AC protocol, and yet easy to implement and deploy [15].

The rest of the article is organized as follows: Section 4.2 presents a description of DACME, the admission control system we selected to implement and test. Section 4.3 offers a brief description of the testbed used for testing. Then, section 4.4 presents the experimental results and discussion. Finally, section 4.5 presents the conclusions of this work, along with future works.

4.2 DACME Overview

In this section we present a brief description of DACME (Distributed Admission Control for MANET Environments). DACME is a distributed admission control module which allows achieving per flow QoS requirements in terms of bandwidth and delay. One of the main advantages of DACME is that it does not impose any specific requirements on MANET nodes, except for IEEE 802.11 and routing support; in fact, DACME agents are only required at the communication endpoints. Since DACME does not have MAC level constraints, it can be implemented in an easy way on all systems that support standard TCP/IP.

The admission control algorithm adopted by DACME relies heavily on bandwidth estimations to decide whether to admit/deny a flow, and uses the values of up to five probes that, combined with the T-Student function, allow obtaining a 95% confidence interval based on which flow acceptance/denial decisions are made. Previous works [5] have shown that this algorithm is highly effective in simulated MANET environments.

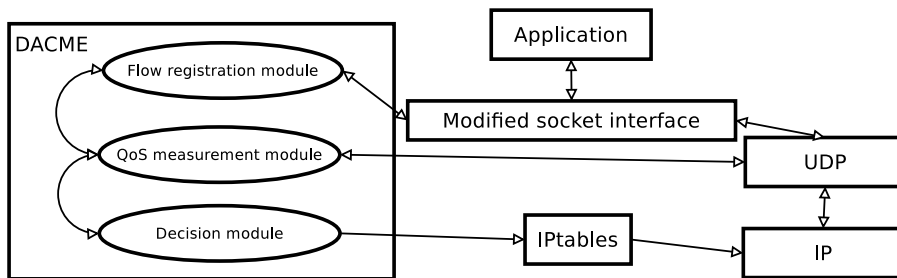


Figure 4.1: DACME diagram illustrating the dependencies among the different elements involved.

To evaluate the effectiveness of DACME in a real environment, we developed an application level library¹ that interacts with both the applications and the kernel to achieve all the functionality required. 4.1 shows the interaction between the different DACME elements.

Initially, the application must register with DACME by using a modified socket interface that also accepts flow QoS specifications (bandwidth required, maximum delay, maximum jitter) as input. This interface, which is part of the library developed, exchanges information with both the operating system (creating a regular socket for communications) and DACME's core. The flow is registered with DACME by including not only the QoS specifications, but also the connection details (source port, destination port, destination IP). In a second step, DACME's QoS measurement module will probe the end-to-end path using the techniques described in [15]. By interacting with the DACME agent at the destination, the DACME agent at the source will gather information that will allow it to make admission control decisions. According to the decision made, this module relies on the iptables tool [9] to dynamically block flows during periods of congestion, and unblocking them when QoS conditions are again met. The IP ToS header field of accepted flows is also modified in order to take advantage of the

IEEE 802.11e Video and Voice medium access categories. Notice that the IEEE 802.11e MAC driver will automatically map the IP ToS values to the four different medium access categories available.

4.3 Testbed Setup

For testing the effectiveness of DACME in a real testbed we used a set of 8 PCs, where 7 of them are low-end Asus EeePC netbooks, and one of them is a desktop computer. All the netbooks have a Ralink RT2860 wireless card, and the desktop system uses a Linksys WUSB600N USB wireless card which employs the Ralink RT2870 chipset. All wireless cards support the IEEE 802.11n draft3 standard, which includes IEEE 802.11e QoS extensions by default. The drivers employed were available in Linux kernel version 2.6.32, allowing us to build a realistic system with all its inherent characteristics and problems.



Figure 4.2: Chain topology used for testing.

The eight stations are wirelessly connected according to a chain topology (see 4.2) to achieve a seven-hop ad-hoc network, which allows us testing with different hop number combinations per flow. The different stations involved in the tests are also interconnected via Ethernet for remote experiment control and, as a return channel, to measure the delay of the UDP packets injected. To avoid high CPU usage, we select different source/destination pairs for the different traffic flows. To introduce variable degrees of congestion in our tests, we also relied on Best Effort traffic flows. Notice that all the video flows share a same link (between terminals E and F) which becomes the network's bottleneck.

To manage the repeatability of the experiments we used Castadiva [29], a Java-based network emulator targeting ad-hoc networks.

In our experiments we vary both the number of best effort traffic flows and QoS (video) traffic flows. To introduce variable degrees of congestion, each best effort traffic flow consists of a 1.5 Mbit/s UDP stream; by varying the number of best effort flows we were able to achieve different channel congestion conditions. With respect to video flows, they consist of synthetic traffic at a rate of 1 Mbit/s (unidirectional). Table 4.1 summarizes the stations acting as sources and destinations of the different video and best effort traffic flows.

Table 4.1: Flow endpoint definition for both Video and Best Effort traffic flows.

Video	Source/Destination	Best Effort	Source/Destination
Flow #1	C → F (3 hops)	Flow #1	A → H (7 hops)
Flow #2	E → H (3 hops)	Flow #2	C → F (3 hops)
Flow #3	G → D (3 hops)	Flow #3	G → B (5 hops)
Flow #4	F → C (3 hops)	Flow #4	E → A (4 hops)

4.4 Experimental results

In this section we present the results of the tests made to assess how best effort traffic affects the stability of video flows. In our experiments we have three concurrent video flows (#1, #2, and #3) which start at random times during the first 20 seconds of the experiment, and then last for 80 seconds. Additionally, for each test, we increase the number of best effort flows from zero to four to assess the impact of congestion on QoS performance. Notice that, for all the results presented, each point corresponds to the mean of

25 independent tests, each lasting 100 seconds. Throughput and delay values include 95% confidence intervals.

4.3 (top) shows the mean throughput per video flow during periods of activity. We can see that bitrate values are maintained close to the maximum if DACME is used. When DACME is not used, we can clearly observe the negative impact of background traffic, drastically affecting the QoS of the video flows. The confidence intervals presented further evidence the goodness of the DACME decision algorithm, showing that bitrate variability associated with this algorithm is much lower compared to the “Without DACME” situation. In particular, we found that the standard deviation when using DACME was never higher than 19% while, when not using DACME, it is never lower than 30%, surpassing 100% in the worst case.

Concerning delay, 4.3 (bottom) shows that the differences between using DACME and not using it are again quite noticeable. In particular, we can see that, when the video flows are managed by DACME, the delay is typically lower than 100ms, being adequate for real-time communication. On the contrary, if we do not use DACME, the mean delay rises up to 700ms, being the lowest value of about 250ms. Similarly to what occurs for throughput, the confidence intervals for the delay are very low if DACME is used, becoming quite wide otherwise.

Overall, the testbed results obtained show that the usage of DACME is of utmost importance to achieve good QoS levels in real environments, which agrees with previous simulation results.

4.5 Conclusions and future work

In this paper we presented the real implementation of DACME, a distributed admission control system for MANET environments

4.5 Conclusions and future work

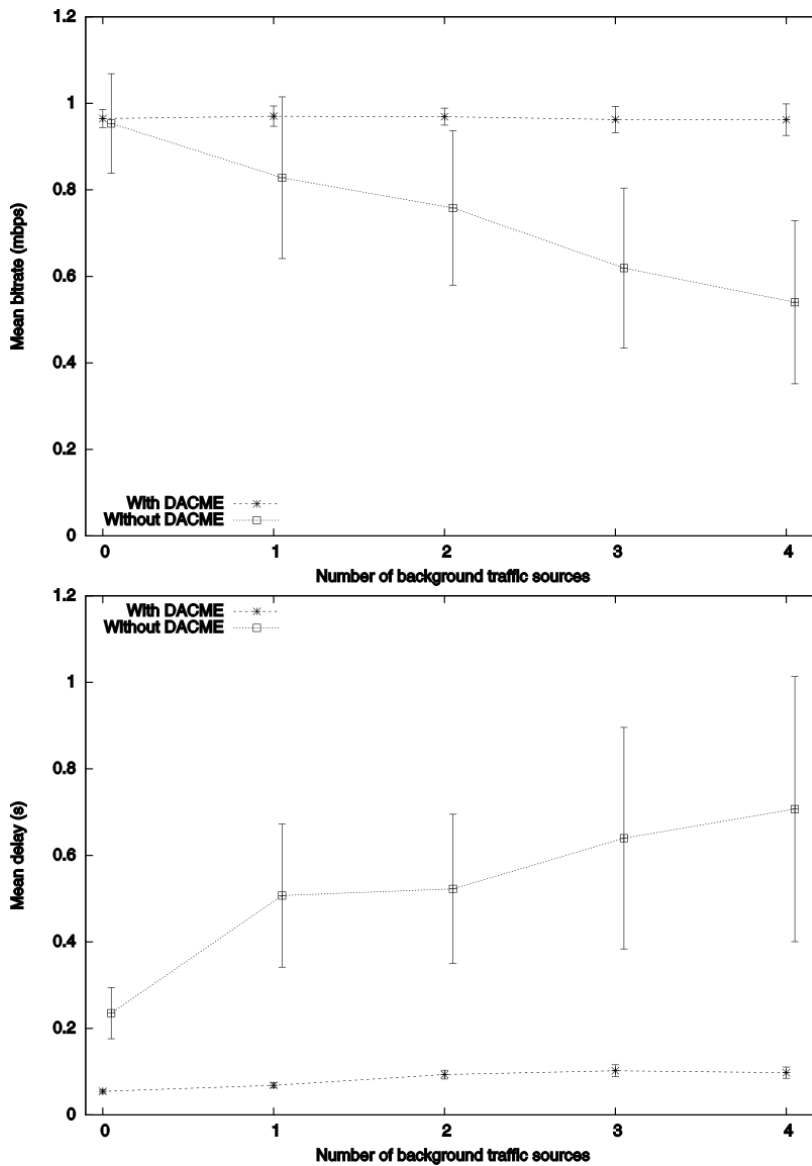


Figure 4.3: Mean bit rate when flows are active (top) and mean end-to-end delay (bottom).

that was earlier validated through simulation. We test its effectiveness in a real testbed using different performance indexes such as the throughput and delay of QoS flows.

Overall, the results presented in this paper clearly show that: (i) traffic differentiation provided by IEEE 802.11e is not enough in real multi-hop ad-hoc networks, and so a distributed admission control like DACME becomes essential; and (ii) the DACME QoS architecture was fully effective in a real testbed, successfully validating the previous simulation results obtained.

As future work we plan to test with scalable video streams, adapting DACME decision algorithms to the multi-level quality characteristics inherent to such streams.

Publication D:
Evaluating the Effectiveness of a QoS
Framework for MANETs in a Real
Testbed

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Evaluating the Effectiveness of a QoS Framework for MANETs in a Real Testbed*. Ad-hoc, Mobile, and Wireless Networks Volume 7363. Pages 221 to 234. 2012

Abstract

Despite all the research efforts in the two previous decades, only a few mobile ad-hoc network (MANET) testbeds have actually been deployed, and even fewer were able to offer Quality of Service (QoS) support. The main problems hindering actual deployment have to do with the distributed effects of mobility, channel contention, and

interference. Using simulation or analytical models, several QoS protocols, architectures and algorithms have been presented with the aim of improving QoS support in MANETs. When attempting to translate these research efforts to real testbeds, the difficulty to represent issues like feasibility in real systems, implementation complexity, node deployment and experiment repeatability have prevented their validation. In this paper we present a real implementation of DACME, the QoS framework we propose for mobile ad-hoc networks, and we test its effectiveness in an IEEE 802.11e enabled testbed. Experimental results show that the developed solution is able to achieve good QoS levels, offering sustained bandwidth levels and bounded delay.

5.1 Introduction

Nowadays, mobile ad hoc networks (MANETs) provide a cheap and infrastructureless form of communication. When combined with an appropriate routing protocol, the IEEE 802.11 standard [26] allows to easily deploy a MANET, which can be very useful in areas where the provision of a central infrastructure is limited or not possible. Typical MANET users share messages and collaborate with each other [17].

Since the IEEE 802.11 standard has been widely used in most wireless LAN environments, the IEEE 802.11e working group [27] proposed an extension to provide QoS support at the MAC level, improving the performance of AP based wireless networks, but also the different networks based on this standard, as in the case of MANETs. The 802.11e extension to the original IEEE 802.11 standard introduces four new traffic categories: Voice, Video, Best Effort, and Background (ordered according to their priority). These four categories provide traffic differentiation by adopting per-category

values for the Contention Window (CW) and the Inter Frame Space (IFS) parameters.

Despite the enhancements that IEEE 802.11e has brought, it is still not enough when facing QoS flow concurrency. In fact, one of the most crucial components of a system attempting to provide QoS guarantees is the Admission Control Module (ACM). This module should be able to estimate the resources of the network and decide when application flows should be admitted or rejected, avoiding to interfere with previously active flows. Unfortunately, this is not an easy task since MANETs are highly dynamic environments, and thus flow admission does not guarantee good QoS conditions throughout time. So, although significant efforts have been done in this area, the solution to the problem is not trivial.

In this paper we develop and evaluate a real implementation (i.e., using a real IEEE 802.11e enabled testbed) of DACME [15], a low power consumption and low complexity end-to-end admission control module. The proposed solution imposes no constraints on the intermediate nodes of the communication other than being able to route packets. Experimental results confirm the goodness of DACME at enhancing QoS support in wireless multi-hop environments. Then, based on the acquired experience, we propose an enhancement to the DACME decision module which offers better adaptability to the network bandwidth fluctuations under wireless interference conditions.

The rest of this paper is organized as follows: in section 5.2 we review the state of the art on admission control solutions for MANETs. Section 5.3 presents a description of DACME, the admission control system we selected to implement and test, along with the proposed enhancement to DACME's decision module. Section 5.4 offers a brief description of the testbed used for testing. Then, section 5.5 presents the experimental results and discussion. Finally, section 5.6 presents the conclusions of this work along with future

works.

5.2 Related works

In the literature we can find several theoretical admission control (AC) algorithms for MANET environments. The main drawback associated with the different solutions we have reviewed is that they have only been tested on simulated environments, and, to the best of our knowledge, none of them has been implemented and tested in a real environment.

According to the guidelines provided on the survey by Hanzo et al. [28], the different AC algorithms available can be divided in two large groups: *routing coupled* and *routing decoupled*. In the first group we can find different algorithms such as ACRMP [22], or MACMAN [33]. All of these AC algorithms require modifying the routing algorithm to support the AC extension. This strategy has some benefits, such as shorter admission times and less overhead of the AC protocol, mainly because they use routing packets to measure the state of the network. However, this first group also presents an important drawback: since they are coupled with a specific routing algorithm, a different routing protocol can not be used without losing the AC module. Another drawback is the strong requirements imposed on nodes, forcing every node in the network to adopt the modified routing protocol to support the AC module. This means that even low power nodes, and nodes which do not require the AC module, will have to dedicate additional resources to support it.

With respect to routing decoupled algorithms, their main advantage is that they allow using any routing protocol for MANET environments. Within the routing decoupled algorithms group, an

additional division can be made between *stateful* and *stateless* protocols. Stateful algorithms save certain information about the state of the links in every node. This strategy allows AC algorithms such as INSIGNIA [32] or MPARC [56] to store information about QoS conditions relative to past flows, and decide, based on this information, whether to accept or reject a flow. Thus, similarly to what occurs for routing coupled protocols, they impose several restrictions on intermediate nodes and require more computing power from these nodes. Within the stateless, routing decoupled protocols group, we can find solutions such as DACME [15], which do not impose any restriction on intermediate nodes since they need not store any information about past flows, nor must they have a high computing power.

In this paper we will develop and test DACME. We chose this proposal since it has been proved to be a powerful and efficient AC protocol, and yet easy to implement and deploy in real systems.

5.3 DACME Overview

In this section we present a brief description of DACME (Distributed Admission Control for MANET Environments) [15]. DACME is a distributed admission control system which allows achieving per flow QoS requirements in terms of bandwidth and delay. One of the main advantages of DACME is that it does not impose any specific requirements on MANET nodes besides the use of IEEE 802.11 and having routing support; in fact, DACME agents are only required at the communication endpoints. Since DACME does not have MAC level constraints, it can be implemented in an easy way on all systems supporting the standard TCP/IP architecture.

5.3.1 Admission decision algorithms meeting bandwidth requirements

DACME learns about the network status using a probe/response strategy. In particular, to achieve an accurate bandwidth estimation, DACME uses a burst of probe packets periodically generated by the source in a back-to-back fashion. These packets arrive to the destination node with an average inter-packet time gap which allows the destination node to make an estimation of the available end-to-end bandwidth. When the destination gathers all the data required, it sends a response packet with the current bandwidth estimation (BM) back to the source. An illustration of this strategy can be found in figure 5.1.

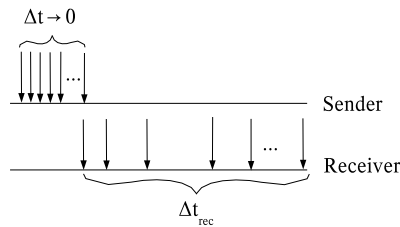


Figure 5.1: Bandwidth estimation strategy.

When receiving a response, the source applies an algorithm to decide whether the flow is accepted or rejected. The admission control algorithm adopted by DACME relies heavily on bandwidth estimations to decide whether to admit or deny a flow. For this reason, in this section we will focus explicitly in this DACME algorithm. In particular, we first describe CIB-DA, the original decision algorithm proposed in [15], and we then propose HRCI-DA, a novel mechanism we have developed based on experimentation, which offers an improved behavior in real environments compared to the former.

Algorithme 5.1 The CIB-DA decision algorithm.

After receiving each BM_i **do** {
 $\mu_i \leftarrow \frac{(i-1) \cdot \mu_{i-1} + BM_i}{i}$, $\sigma_i \leftarrow \sqrt{\frac{(i-2) \cdot (\sigma_{i-1})^2 + (BM_i - \mu_i)^2}{i-1}}$
 if $(\mu_i - t_{i-1,0.95} \frac{\sigma_i}{\sqrt{i}} > B_R)$
 then $Flag(BW) \leftarrow 1$
 else if $(\mu_i + t_{i-1,0.95} \frac{\sigma_i}{\sqrt{i}} < B_R)$
 then $Flag(BW) \leftarrow 0$
 else if $(i < 5)$
 then send a new probe }

5.3.1.1 CIB-DA: Confidence Interval Based Decision Algorithm

The CIB-DA algorithm uses the values of up to five probes to obtain a 95% confidence interval for the available bandwidth value, based on which flow acceptance/denial decisions are made. Algorithm 5.1 shows the pseudo-code that describes the behavior of the CIB-DA algorithm. Previous works [15] have shown that this algorithm is highly effective in simulated MANET environments.

CIB-DA is executed every time a probe reply is received. Decisions are based on statistical confidence levels; therefore, i refers to the current iteration, $t_{i-1,0.95}$ to a Student's t-distribution with $i - 1$ degrees of freedom, and for a confidence level of 95%. Parameter B_R refers to the bandwidth required by the application, while BM refers to the bandwidth measurement explained previously. If the application is solely bandwidth constrained, the value of the bandwidth flag - $Flag(BW)$ - will determine whether the QoS flow can be accepted. If the application has delay requirements as well, the value of the delay flag is also considered. These issues are addressed

in later sections.

5.3.1.2 HRCI-DA: Hybrid Range/Confidence Interval Decision Algorithm

Contrarily to what occurs in simulations, lots of problems exist in real testbeds, not only at the transmission level (e.g. interferences, packet loss), but also at the application, kernel, and hardware levels. Examples of effects occurring at these levels include (but are not limited to) CPU Usage, RAM paging, time measurement delays, and loss of synchronization.

Focusing on the interferences problem, one of the main differences between simulation and real testbed experiments has to do with the wireless channel. While in simulation experiments wireless channels are free from external interferences, in testbeds this is rarely true. In fact, in real environments, our implementation of DACME using the CIB-DA algorithm exhibits significant interferences, which caused estimated bandwidth to experience frequent and drastic fluctuations. This impeded obtaining low confidence intervals in most situations since the CIB-DA algorithm used a t-Student function with only a few degrees of freedom for calculating those intervals. Such large confidence intervals provoked that most flows were not accepted, or decisions could not be made even when the minimum bandwidth values measured were higher than the demanded ones.

To avoid this problem we proposed the HRCI-DA algorithm, an improvement to the CIB-DA measurement algorithm. The main goal of HRCI-DA is to bound the interval used to make flow acceptance/denial decisions, never allowing it to become higher than the half range, that is, half the difference between the maximum and

Algorithme 5.2 The HRCI-DA decision algorithm.

After receiving each BM_i **do** {
 $range \leftarrow \max(\{BM\}) - \min(\{BM\})$
 $\mu_i \leftarrow \frac{(i-1) \cdot \mu_{i-1} + BM_i}{i}$, $\sigma_i \leftarrow \sqrt{\frac{(i-2) \cdot (\sigma_{i-1})^2 + (BM_i - \mu_i)^2}{i-1}}$
 find the unbiased bandwidth estimator $v_{p,i}$
if $(\frac{range}{2} > t_{i-1,0.95} \frac{\sigma_i}{\sqrt{i}})$ **then**
 $(\hat{BW}_{low}, \hat{BW}_{high}) \leftarrow (\mu_i - t_{i-1,0.95} \frac{\sigma_i}{\sqrt{i}}, \mu_i + t_{i-1,0.95} \frac{\sigma_i}{\sqrt{i}})$
else $(\hat{BW}_{low}, \hat{BW}_{high}) \leftarrow (\mu_i - \frac{range}{2}, \mu_i + \frac{range}{2})$
if $(\hat{BW}_{low} > B_R)$ **then** $Flag(BW) \leftarrow 1$
else if $(\hat{BW}_{high} < B_R)$ **then** $Flag(BW) \leftarrow 0$
else if $(i < 5)$ **then** send a new probe }

minimum measurements. Algorithm 5.2 shows the pseudo-code for HRCI-DA.

In this algorithm two different intervals are used to determine the value of the bandwidth flag - $Flag(BW)$ -, being one based on the range of the values, and the other based on confidence intervals. The former is used whenever the values produced by the latter strategy are excessive.

5.3.2 Admission decision algorithms meeting delay requirements

For delay estimations, DACME employs a ping-pong method without any delay between consecutive request/response patterns. At the sender node, the measured RTT of the end-to-end path is used to estimate network delay. To obtain the delay value and determine whether a new flow is going to be accepted, an adjustment function is used; this function is described in [15], and its main pur-

pose is to make short-term measurements match long-term ones. Only when both delay and bandwidth restrictions are achieved can a flow be accepted, being blocked otherwise. An illustration of the basic delay estimation strategy is shown in figure 5.2.

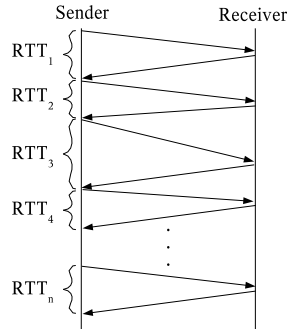


Figure 5.2: Delay estimation method.

5.3.3 Implementation details

To evaluate the effectiveness of DACME in a real environment, we developed an application level library¹ that interacts with both the applications and the kernel to achieve all the required functionality. Figure 5.3 shows the interaction between the different DACME elements.

Initially the application must register with DACME by using a modified socket interface that also accepts flow QoS specifications (bandwidth required, maximum delay, maximum jitter) as input. This interface, which is part of the developed library, exchanges information with both the operating system (creating a regular

¹Freely available at: <http://dacme.sourceforge.net/>

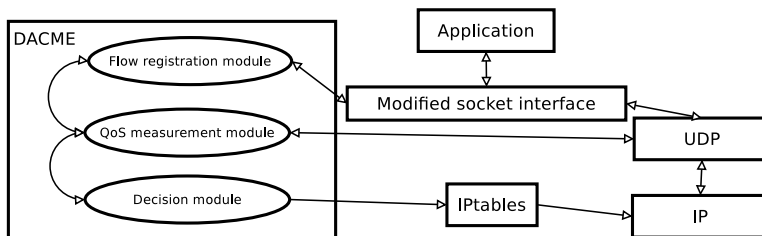


Figure 5.3: DACME diagram illustrating the dependencies among the different elements involved.

socket for communications) and DACME's core. The flow is registered with DACME by including not only the QoS specifications, but also the connection details (source port, destination port, destination IP). In a second step, DACME's QoS measurement module will probe the end-to-end path using the techniques described above. By interacting with the DACME agent at the destination, the DACME agent at the source will gather information that will allow it to make admission control decisions using any of the bandwidth-based decision algorithms presented in section 5.3.1, and the delay-based decision algorithms presented in section 5.3.2. According to the decision made, this module relies on the *iptables* tool [9] to dynamically block flows during periods of congestion, and unblocking them when QoS conditions are again met. The IP ToS header field of accepted flows is also modified in order to take advantage of the IEEE 802.11e Video and Voice medium access categories. Notice that the IEEE 802.11e MAC driver will automatically map the IP ToS values to the four different medium access categories available.

5.4 Testbed setup

For testing the effectiveness of DACME in a real testbed we used a set of Asus EeePC netbooks, regular laptops, and a desktop computer. All the netbooks have a Ralink RT2860 wireless card, and the laptop and desktop systems use a Linksys WUSB600N USB wireless card which employs the Ralink RT2870 chipset. All wireless cards support the IEEE 802.11n draft3 standard, which includes IEEE 802.11e QoS extensions by default. The drivers employed were available in Linux kernel version 2.6.32, allowing us to build a realistic system with all its inherent characteristics and problems.

The stations are wirelessly distributed and connected to achieve a seven-hop ad-hoc network, which allows us testing with different hop number combinations per flow. The different stations involved in the tests are also interconnected via Ethernet for remote experiment control, and as a return channel to measure the delay of the UDP packets injected. To avoid high CPU usage, we select different source/destination pairs for the different traffic flows. To introduce variable degrees of congestion in our tests, we also relied on Best Effort traffic flows, and all the video flows share a same link which becomes the network's bottleneck.

To manage the repeatability of the experiments we used Castadiva [29], which we extended to provide DACME compatibility. Castadiva allows us to automate large sets of experiments and collect all the statistics required.

In our experiments we vary both the number of best effort traffic flows and QoS (video) traffic flows. To introduce variable degrees of congestion, each best effort traffic flow consists of a 1.5 Mbit/s UDP stream; by varying the number of best effort flows we were able to achieve different channel congestion levels. With respect to video flows, they consisted of synthetic traffic at a rate of 1

5.5 Experimental Results

Mbit/s (unidirectional) demanding to DACME a maximum end-to-end delay of 300ms. Table 5.1 summarizes the number of hops for each flow. Additionally, one of the links is shared by all traffic flows to aggravate the problem of congestion. Notice that all hop count values are representative of typical MANET studies.

Table 5.1: Flow endpoint definition for both video and best effort traffic flows.

<i>Video</i>	<i>Number of hops</i>	<i>Best effort</i>	<i>Number of hops</i>
Flow #1	3 hops	Flow #1	7 hops
Flow #2	3 hops	Flow #2	3 hops
Flow #3	3 hops	Flow #3	5 hops
Flow #4	3 hops	Flow #4	4 hops

5.5 Experimental Results

In this section we perform a detailed evaluation to assess the correct functionality of our implementation of DACME, as well as the improvements introduced by both DACME AC algorithms (CIB-DA and HRCI-DA). With this goal we created three scenarios: in the first one we varied congestion by increasing the number of best-effort flows (background traffic), in the second one we increased the number of competing QoS flows, and, in the last one, we varied the maximum delay restriction for QoS flows. Our goal was to study the QoS stability of DACME flows in a real environment. Thus, for each test, the performance parameters under analysis were: throughput, delay, total activity time per-flow, and mean number of DACME on/off state transitions. This last parameter is interesting as the less a flow switches it's state, the more stable the flow is for an end-user.

5.5.1 Varying the number of Best Effort flows

In this first set of experiments we study how best effort traffic affects the stability of video flows. In our experiments we have three concurrent video flows which start at random times during the first 20 seconds of the experiment, and then they last for 80 additional seconds. Also, for each test, we increase the number of best effort flows from zero to four to assess the impact of congestion on QoS performance. Notice that, for all the presented results, each measurement corresponds to the mean value of 25 independent tests. Throughput and delay values include 95% confidence intervals.

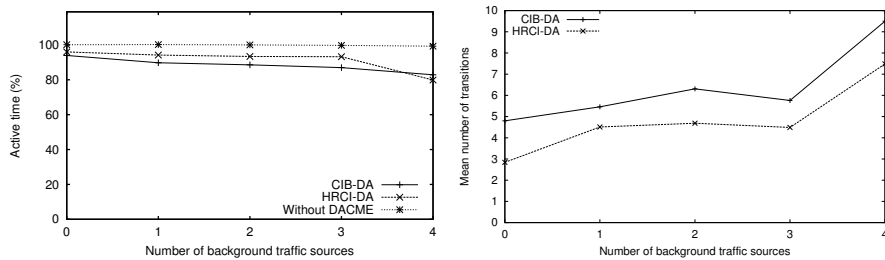


Figure 5.4: Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the number of background traffic sources.

Figure 5.4 (left) shows the mean time each QoS flow is active. Notice that, if we do not use DACME, the activity time is always the maximum. The usage of DACME implies that, during certain periods, the flow can not be accepted since QoS requirements are not met; however, we find that, on average, the activity periods are maintained high, as desired. Focusing on the differences between using and not using DACME, we can see, significant benefits in terms of both bandwidth and delay values.

Figure 5.5 (left) shows the mean throughput per video flow during periods of activity. We can see that bitrate values are maintained

close to the maximum if DACME is used (for both admission control algorithms). When DACME is not used, we can clearly observe the negative impact of background traffic, drastically affecting the QoS of the video flows. The confidence intervals presented further evidence the goodness of both DACME decision algorithms, showing that bitrate variability associated with these algorithms is much lower compared to the “Without DACME” situation. In particular, we found that the standard deviation when using DACME was never higher than 19% while, when not using DACME, it is never lower than 30%, surpassing 100% in the worst case.

Concerning delay, figure 5.5 (right) shows that the differences between using DACME and not using it are again quite noticeable. In particular, we can see that, when the video flows are managed by DACME, the delay is typically lower than 100ms, therefore being adequate for real-time communication. On the contrary, if we do not use DACME, the mean delay rises up to 700ms, being the lowest value of about 250ms. Similarly to what occurs for throughput, the confidence intervals for the delay are very low if DACME is used, becoming quite high otherwise.

If we look at the differences between CIB-DA and HRCI-DA, we find that the former is clearly more restrictive, typically introducing more flow blockage in order to meet the QoS requirements. This can be observed in Figure 5.4 (right), which shows the mean number of state transitions per flow, that is, transitions from active to blocked state, or the opposite. Since DACME performs periodic bandwidth measurements, it can block flows during ongoing communication as soon as QoS loss is detected. Although these dynamic decisions are mandatory to handle the effects of mobility, it is important to maintain this value as low as possible. Results show that our HRCI-DA algorithm offers significant improvements compared to CIB-DA, reducing the mean number of state transitions by up to 40%, and making communication more fluid, while

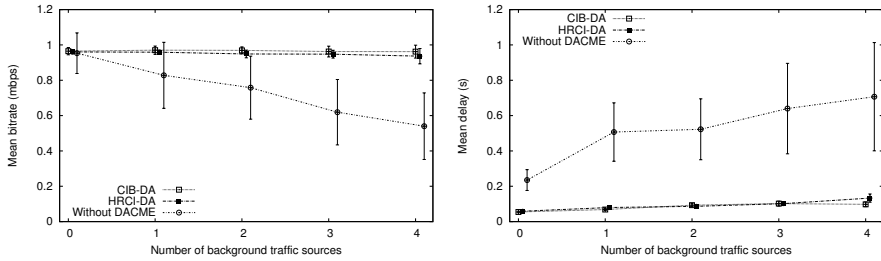


Figure 5.5: Mean values for throughput (left) and end-to-end delay (right) when varying the number of background traffic sources.

not being differentiable in terms of throughput or delay from CIB-DA.

5.5.2 Varying the number of Video Flows

The goal of this second set of experiments is to assess the performance of our implementation of DACME when handling a variable number of QoS flows, as well as analyzing the interactions between these flows. With this purpose our experimental settings are similar to those of the previous section, but we now fix the number of best effort flows to three, while increasing the number of video flows from one to four.

Figure 5.6 (left) shows the results of the active time per flow. Similarly to the previous section, we find that the percentage is quite high for both DACME algorithms, and that increasing the number of video sources does not cause a proportional decrease in terms of activity time. This is mostly due to the distributed nature of wireless channel access in MANETs.

While being able to maintain a high activity time, in terms of throughput, figure 5.7 (left) shows that both DACME AC algorithms allow achieving a similar throughput (nearly 1 Mbit/s)

5.5 Experimental Results

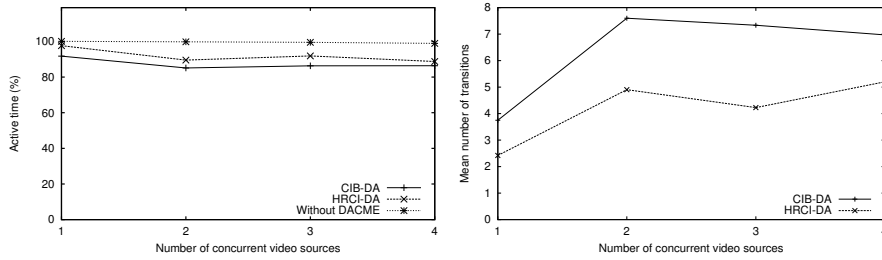


Figure 5.6: Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the number of concurrent video sources.

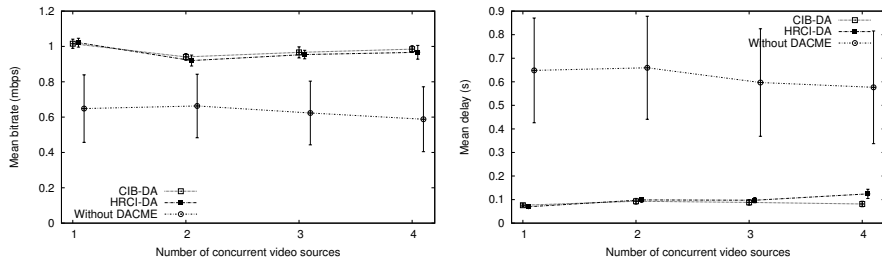


Figure 5.7: Mean values for throughput (left) and end-to-end delay (right) when varying the number of concurrent video sources.

while, when DACME is not used, this value drops to about 0.6 Mbit/s. As in the previous set of tests, the confidence interval obtained is much lower when we use DACME, meaning that variability is strongly reduced compared to the “Without DACME” case. This set of experiments allows concluding that, even when using IEEE 802.11e to achieve traffic differentiation at the MAC layer, the Video traffic category at the MAC layer is still highly affected by Best Effort traffic interferences; thus, although IEEE 802.11e allows some differentiation between the different traffic categories, it is not powerful enough to provide, by itself, full QoS guarantees

in MANET scenarios.

Focusing on the delay (see figure 5.7, right), the behavior is similar to the previous tests. Again, the delay experienced by QoS traffic becomes excessive when DACME is not used. Additionally, we find that, when using either DACME decision algorithm, the delay experienced by the video flows is maintained low, and mostly immune to the increase of video sources.

Comparing both decision algorithms, we find that HRCI-DA offers better results in terms of both activity time (figure 5.6, left) and mean number of state transitions (figure 5.6, right). In fact, the latter experiences a reduction of up to 30%, which shows the effectiveness of HRCI-DA at improving video streaming stability in real environments compared to its predecessor.

5.5.3 Varying the maximum delay restriction

In this third and last set of experiments our goal is to validate our implementation of DACME when varying the maximum delay allowed for the video flows to determine the degree of compliance achieved. With this aim we fix the number of Video and Best Effort flows to three each, and we vary the maximum delay restrictions from 100 to 600 ms, again comparing both DACME AC algorithms against a solution where DACME is not used.

Figure 5.9 (left) shows that DACME is able to sustain the bitrate values at near-optimum levels and with little variability, contrarily to the “Without DACME” situation.

With respect to the delay experienced (Figure 5.9, right), we find that the mean delay values are very low when we use DACME compared with the “Without DACME” situation. If we look deeper into the delay restriction accomplishments, for the lowest value (100 ms), about 70% of the packets comply with the maximum

5.5 Experimental Results

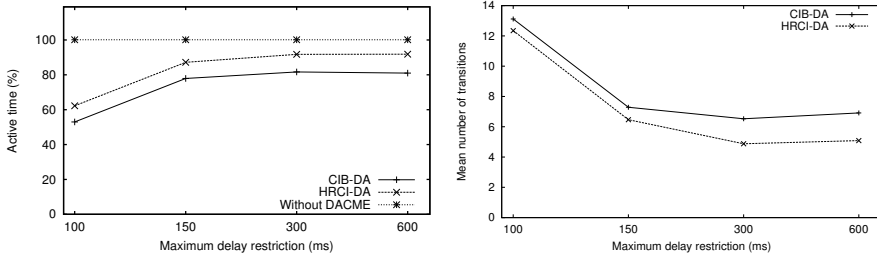


Figure 5.8: Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the maximum delay restriction.

delay restriction, while for 150ms the level of delay compliance is of 83%, growing to more than 94% for the last two cases (300 and 600 ms).

Comparing both DACME algorithms, Figure 5.8 (left) shows the mean time of activity when increasing the maximum delay allowed. We find that low delay requirements have a significant impact on activity time, reducing it up to 46% for both AC strategies. Again, HRCI-DA is able to improve the mean time of activity achieved by CIB-DA, achieving an increase of up to 19%. For this metric, the “Without DACME” values remain constant since the different delay restrictions are only meaningful within the scope of DACME.

Varying the maximum delay restriction has also a significant impact in terms of the mean number of state transitions, as shown in Figure 5.8 (right). Again HRCI-DA shows a better behavior, in this case by reducing the number of state transitions involved. It is also worth noticing that the main differences detected occur when increasing the maximum delay restriction from 100 to 150 ms, being the overall behavior mostly maintained afterward. This occurs because most of the delay values measured are in the 100-150 ms range.

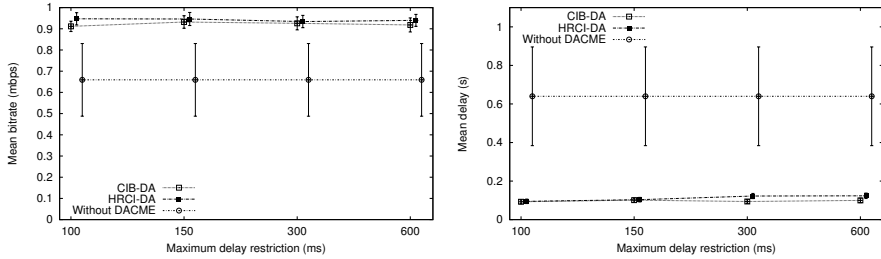


Figure 5.9: Mean values for throughput (left) and end-to-end delay (right) when varying the maximum delay restriction.

Overall, the results presented in this section validate the effectiveness of DACME in real testbeds, and evidence the improvements introduced by the HRCI-DA decision algorithm compared to its predecessor (CIB-DA) in real environments. Our HRCI-DA algorithm is able to increase the overall activity time and reduce the number of transitions, while maintaining good QoS values in terms of both throughput and delay. Hence, we can conclude that our DACME implementation mostly retains the QoS properties inferred based on simulation results, although some adjustments can help at further boosting performance in real environments.

5.6 Conclusions and future work

In this paper we validated a real implementation of a distributed admission control system for MANETs that was previously evaluated through simulation. We test its effectiveness in a real testbed using different performance indexes such as throughput, delay, and total time of activity.

To cope with bandwidth estimation accuracy problems occurring in real environments, we proposed an enhanced decision algorithm

(HRCI-DA) for the admission control module that offers significant performance improvements compared to the previous version (CIB-DA). HRCI-DA reduces the number of on/off state transitions for QoS flows, and improves the total activity times, while also maintaining good throughput and delay values.

Overall, the results presented in this paper clearly show that: (i) traffic differentiation provided by IEEE 802.11e is not enough in real multi-hop ad-hoc networks, and so a distributed admission control like DACME becomes essential; (ii) the DACME QoS architecture was fully effective in a real testbed, successfully validating the previous simulation results obtained; (iii) the proposed HRCI-DA algorithm improves the original one by providing greater stability to QoS flows, increasing their total activity time and reducing the total number of on/off state transitions; and (iv) despite the greater number of active QoS-flows, bandwidth and delay values are maintained or even improved by HRCI-DA compared to CIB-DA.

As future work we plan to test our implementation with scalable video streams, adapting DACME decision algorithms to the multi-level quality characteristics inherent to such streams, thus offering the possibility for adaptive real-time multimedia traffic in MANETs.

**Publication E:
Evaluating H.265 Real-Time Video
Flooding Quality in Highway V2V
Environments**

Alvaro Torres, Pablo Piñol, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Evaluating H.265 Real-Time Video Flooding Quality in Highway V2V Environments*. Wireless Communications and Networking Conference (WCNC), 2014. Pages 2716 to 2721. April 2014

Abstract

Video transmission over VANETs is an extremely difficult task not only due to the high bandwidth requirements, but also due to

typical VANET characteristics such as signal attenuation, packet losses, high relative speeds and fast topology changes.

In future scenarios, vehicles will provide other vehicles with information about accidents or congestion on the road, and in these cases offering visual information can be a really valuable resource for both drivers and traffic authorities. Hence, achieving an efficient transmission is critical to maximize the user-perceived quality. In this paper we evaluate solutions that combine different flooding techniques, and different video codecs to assess the effectiveness of long-distance real-time video streaming. In particular, we will compare the most effective video coding standard available (H.264) with the upcoming H.265 codec in terms of both frame loss and PSNR.

6.1 Introduction

Vehicular ad-hoc networks (VANETs) are receiving a lot of attention from vehicle manufacturers since they promote smarter, cleaner, and safer vehicles. In a near future it is expected that vehicular networks will be as extended as mobile phones are nowadays. Vehicles will be equipped with different sensors which can provide useful information to other drivers or traffic authorities. One of the most useful sources of information a vehicle can provide is the live video of an accident, not only allowing the emergency services to know in advance the exact status of the accident, but also to help other vehicles to decide whether to change their current route.

VANETs provide one of the most difficult environments to achieve a good transmission quality since this type of networks involves high relative speeds, which cause short connection times and transmission problems such as the Doppler effect. In addition, bandwidth

is typically very limited, thus becoming one of the worst case scenarios for real-time video transmission. With these constraints it is necessary to check whether it is possible to achieve real-time video transmission with current broadcasting techniques, and, if so, evaluate the final quality of the obtained video.

The recent approval of the new H.265 video compression standard [2], which intends to replace the widely used and well-known H.264 standard [1], provides a new opportunity for real-time video transmission in critical contexts. The new standard, which outperforms the old one achieving the same video quality with only 50% of the bit-rate [31], may be one of the key points in the achievement of real-time video transmission in vehicular networks, so it is also necessary to evaluate the differences between both of them.

The rest of the paper is structured as follows. In section 6.2, we review the state of the art in terms of both flooding in wireless networks and video transmission over VANETs. In section 6.3, details are provided about the scenario characteristics and the intended video transmission mechanism. Afterward, section 6.4 presents the tools adopted to simulate the proposed scenario. Section 6.5 presents the obtained results and finally, in section 6.6, we summarize the conclusions obtained and discuss future work.

6.2 Related Work

In the existing literature several flooding mechanisms have been presented, although the majority of them are intended for MANETs.

Due to the nature of the IEEE 802.11 protocol, which provides a contention-based broadcast mechanism, flooding protocols are focused on avoiding the broadcast storm problem. Yu-Chee Tseng et al. [39] presented the most basic algorithms to solve this problem.

The same author published [52] an improved version of the same algorithms by adding adaptive conditions to further reduce the broadcast storm problem. More specifically, the authors present the basis of flooding in MANET environments. In [39] they present the basic versions of some flooding mechanisms such as the Counter-Based Scheme, the Distance-Based Scheme, or the Location-Based Scheme. Based on these algorithms, in [52] they slightly improve these flooding schemes by adding information about the environment to decide whether to rebroadcast a packet.

Martínez et al. [34] proposed another flooding mechanism that takes into account the particularities of VANETs, following the guidelines of the Distance-Based mechanism; in particular, they tweak it to provide a fast dissemination of accident alerts in urban scenarios by using information such as the town layout to achieve a smarter flooding. For highway scenarios the algorithm does not significantly differ from the Distance-Based approach.

The field of video transmission over VANETs has been studied by several authors, although the main purpose of most articles on video transmission is entertainment, so the video is streamed from Road Side Units (RSU) into the vehicular network. Meng Guo et al. [25] presented several scenarios where the live video streaming between vehicles, and from vehicles to RSU, is both feasible and desirable.

F. Soldo et al. [47] presented the SUV protocol, a distributed solution to disseminate video streams in VANETs, although a special MAC layer is required to support TDMA scheduling, which avoids its implementation on actual IEEE 802.11p devices.

Overall, although the aforementioned works provide a good number of statistics, none of them presents actual video quality results such as PSNR, being unaware of decoding problems such as the interdependence between frames.

A first approach towards a proper simulation environment capable of representing real-time video transmission was proposed in [41], where authors present a simulation platform and PSNR results, following the guidelines presented in [45], thereby providing a useful approach on how to accurately simulate a video transmission.

In this paper we present a performance evaluation of different real-time video flooding methods, evaluating the error resilience in terms of frame loss and PSNR for both the standard H.264 codec, and the new H.265 standard. To the best of our knowledge it is the first time that the new H.265 video codec has been evaluated in lossy VANET environments.

6.3 Overview of the video delivery strategy

Based on the current state of the art, we have implemented two video flooding strategies focusing on some desirable characteristics:

1. Achieve a high percentage of delivered packets.
2. Promote a fast message propagation.
3. Make the implementation feasible in a real environment.

To meet these goals, our flooding mechanism implementation is located at the application layer, using UDP as the transport mechanism. This choice avoids the possibility of aborting the transmission of a message when it's scheduled for transmission at lower layers, and also impedes us from obtaining MAC-layer statistics, such as the state of the wireless channel or the number of collisions,

while processing the incoming messages. However, the implementation does not depend on any tweaked MAC layer, and it can be deployed on real existing devices.

The two flooding strategies implemented are the Counter-Based strategy, and the Distance-Based strategy. Below we provide an overview of them.

6.3.1 Counter-Based strategy

The parameters employed by this algorithm are two. The first one is the number of copies (C) that a node should hear to stop rebroadcasting a message, and the second one is the maximum time ($MaxTime$) to rebroadcast.

The exact behavior of our implementation is the following:

1. When a new message arrives, initialize the *seen* counter to 0. If the received message was heard previously, increment the local *seen* counter.
2. Wait a random time between 0 and $MaxTime$.
3. If $seen < C$, rebroadcast the message, go back to 2 and wait for $MaxTime$. Otherwise ($seen \geq C$), discard the message.

6.3.2 Distance-Based strategy

Our implementation is based on the Counter-Based strategy, adding a new parameter, the minimum distance ($MinDistance$) to rebroadcast a message. If the distance between a node and the origin or a rebroadcaster node is greater than the minimum distance defined, it will wait for a period between 0 and $MaxTime$. If this distance is less than $MinDistance$ it will wait for $MaxTime$.

6.4 Simulation Environment

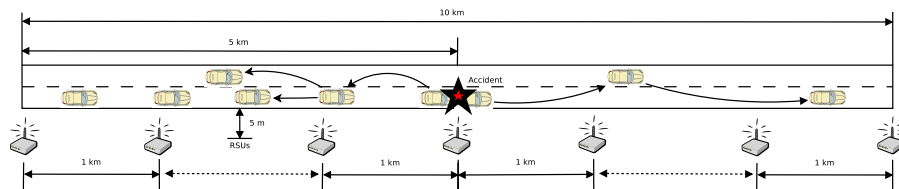


Figure 6.1: Overview of the simulated scenario.

For this implementation it is assumed that every vehicle is equipped with a GPS system, that the messages are marked with the position of the original sender and, if applicable, with the position of the rebroadcasting node.

The exact behavior of our implementation is the following:

1. When a new message arrives initialize the *seen* counter to 0. If the received message was heard previously, increment the local *seen* counter.
2. Obtain the distances to the original sending node (*OriginalDistance*) and the re-broadcaster node (*RebroadcasterDistance*).
3. Wait for a time between 0 and *MaxTime*, where the time is inversely proportional to the minimum of the previously obtained distances.
4. If $seen < C$, rebroadcast the message, go back to point number 2, and wait for *MaxTime*. Otherwise ($seen \geq C$), discard the message.

6.4 Simulation Environment

The simulation environment consists of three main components: SUMO [14], OMNeT++ [6], and INET [3].

To achieve a realistic vehicle mobility we employ the well known open-source vehicular traffic simulator SUMO (Simulation of Urban MObility), which runs coupled with the event-driven OM-NeT++ simulator. To simulate the wireless environment we use the INET package that provides an implementation of the IEEE 802.11p standard, as well as higher application levels such as TCP and UDP.

The transmission range in the INET framework is not defined as a fixed distance. Instead, it requires tuning different parameters such as the frequency or the level of attenuation with distance. To achieve a proper level of similarity with reality, we employed the parameters proposed by Báguena et al. [11].

Since highways are a very specific scenario due to the high speeds, we have reduced the default SUMO step time from 1s to 0.1s. This provides a greater resemblance with real-life behavior since vehicles only move for a distance of 3 meters on each mobility simulation step. Another issue handled by SUMO is the speed of the different nodes. We have created four types of vehicles with different probabilities of appearance, and different speeds in order to achieve realistic vehicle behaviors in terms of overtaking, thereby avoiding large vehicle queues on highways.

Concerning the scenario itself, the highway consists of a one-way two-lane 10km long straight line. A set of 11 Road Side Units (RSU) have been placed along the road every kilometer. The distance from RSUs to the road is 5 meters. Figure 6.1 shows our selected scenario.

In our experiments RSUs do not resend any messages, instead, they merely act as traffic sinks. Our goal is to determine the performance of video delivery at RSUs at different distances from the source vehicle.

The vehicle acting as a video source is stopped when it arrives at

the middle of the scenario (fifth kilometer) and starts transmitting a video sequence simulating an accident which involves an emergency video stream.

The actual transmitted video is the CIF version of the “Highway” video sequence, which contains 2000 frames [10]. The video has been encoded with the H.264 and the H.265 reference codecs, both set to a quality level of about 37.9 dB. (37.95 dB and 37.87 dB respectively). This quality level corresponds to data rates of 347 kbit/s and 283 kbit/s for H.264, and H.265 respectively. Both encoders were configured to create RTP packetized streams, so they have the same overhead. To avoid large dependencies between frames and trying to achieve a better packet loss resilience, key-frames are generated every two seconds.

Additionally, to add real-time constraints, we assume a video buffer of 1s, meaning that a vehicle will start the video playback 1s after the arrival of the first packet, and will discard every packet received beyond this jitter threshold.

6.5 Results

This section is structured as follows. First we make an analysis of the different parameters for both flooding algorithms, distance and counter-based, at different vehicle densities. After that, and focusing on the video codecs, we will compare both codecs in terms of PSNR and frame loss using the best flooding algorithms according to the previous analysis.

Every single simulation configuration was repeated 25 times, so the points shown in the figures are the mean of the obtained values.

6.5.1 Tuning the flooding algorithms

The main objective of a good flooding algorithm is to reduce the number of transmissions to avoid the broadcast storm problem, while achieving a good packet delivery ratio.

Concerning the first algorithm (Counter-Based) we tested three levels for each parameter, achieving a total of nine configurations. In this case, the C value denotes the additional number of times that a host should hear the message to stop rebroadcasting it, and $MaxTime$ stands for the maximum time (in ms) that a node waits when attempting to rebroadcast a message. The second algorithm (Distance-Based) adds another parameter, $MinDistance$. As explained before, if the receiver node is closer than $MinDistance$ to the sender node, it will wait for $MaxTime$ to rebroadcast the message. If the distance is higher, the node will rebroadcast earlier when the distance to the sender node is higher, to maximize the useful transmissions.

The actual set of values employed for these simulations is the following:

- $C = \{1, 2, 3\}$
- $MaxTime = \{50ms, 200ms, 333ms\}$
- $MinDistance = \{50m, 250m, 400m\}$

Figure 6.2 shows the performance of both algorithms when the vehicle inter-arrival time follows an exponential distribution with a mean of $1s$ (high-density scenario).

The best configuration for the Counter-based solution is, without any doubt, the one with $C = 1$ and $MaxTime = 333ms$. This allows achieving a mean delivery rate of 72.23% at the RSUs located five kilometers away, and of 82.63% at the ones located three

6.5 Results

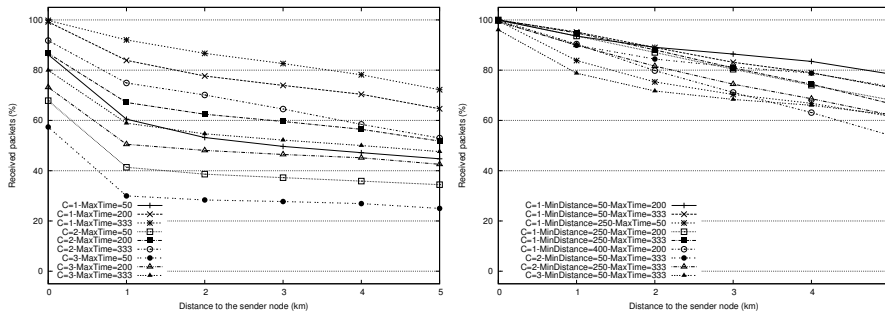


Figure 6.2: Counter-Based (left) vs. Distance-Based (right). Vehicle arrival rate of 1s.

kilometers away. This configuration clearly outperforms the other ones by lowering the number of required transmissions. Increasing the number of copies a node should receive to stop rebroadcasting significantly reduces the effectiveness of the flooding process since it just adds more interference to the medium. The same happens when we reduce the *MaxTime* parameter, as nodes have more probability to collide when gaining access to the medium.

Looking at the right part of figure 6.2, we can see the results for the Distance-based algorithm. Since the number of different configurations for this algorithm is too high to be shown in a graph, we selected the nine best performing configurations. The best arrival rate at five kilometers distance is obtained by the configuration: $C = 1$, $MinDistance = 50$ and $MaxTime = 200ms$, achieving a packet delivery ratio of 78.5% at five kilometers, and of 86.45% at the third kilometer.

If we compare both solutions, the Distance-Based one is able to deliver more packets since it provides a better diffusion of the information, prioritizing the furthest nodes and making each transmission more profitable.

Figures 6.3 and 6.4, which provide the data for vehicle inter-arrival

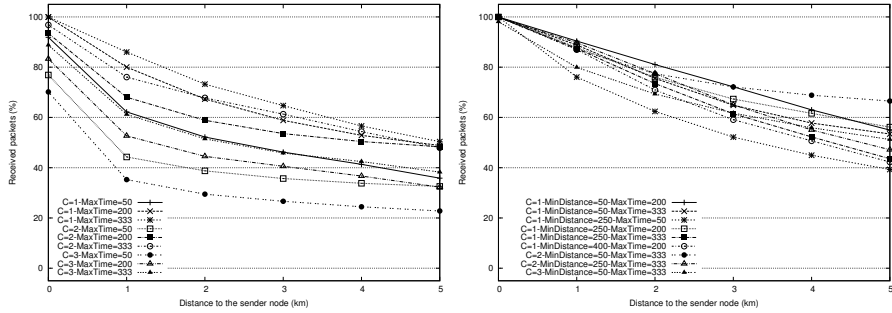


Figure 6.3: Counter-Based (left) vs. Distance-Based (right). Vehicle arrival rate of 2s.

times of 2s (medium-density scenario) and 3s (low-density scenario), show that, for both densities, the best Counter-Based configuration is the same as in a high-density scenario but, due to the lower number of vehicles, the arrival rate is clearly reduced, achieving only a 50.34% of delivered packets at a five kilometers distance in the medium-density scenario, and a 38.11% in the low-density scenario.

Taking into account the Distance-Based algorithms, the right part of figures 6.3 and 6.4 shows that the previous Distance-Based winner configuration is not the best one for medium-density and low-density scenarios. The best configuration for these scenarios is $C = 2$, $MinDistance = 50$, and $MaxTime = 333ms$, achieving a delivery rate at the fifth kilometer of 66.53% and 46.32% for medium-density and low-density scenarios, respectively. Such differences in terms of optimal configuration are due to collisions near the source vehicle. On the right hand side of figure 6.3, focusing on the delivery rates from 0km to 3km, the best configuration is $C = 1$, $MinDistance = 50$, and $MaxTime = 200ms$, while from the third kilometer to the fifth, the best configuration is the one referred above. The main difference between these two configura-

6.5 Results

tions is the C parameter as it stands for the number of times a message should be heard before stopping its rebroadcasting.

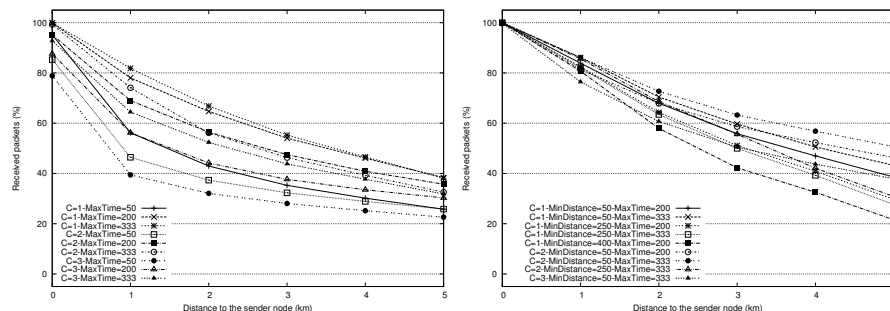


Figure 6.4: Counter-Based (left) vs. Distance-Based (right). Vehicle arrival rate of 3s.

6.5.2 Evaluating the best codec / flooding algorithm combination

With the purpose of evaluating the actual received video quality we have selected the three best flooding configurations analyzed in previous section.

Figure 6.5 shows the ratio of packets received for different vehicle arrival rates depending on the selected codec. The three graphs clearly show the differences between flooding backwards or forwards. When flooding backwards (from kilometer 0 to 5) the delivery ratio increases as the vehicles start creating a traffic jam, provoking a higher density of vehicles which allows using more vehicles as relays. The opposite process (forward flooding) is a harder task since, when vehicles surpass the accident position, they continue driving at a high speed, therefore creating several disjoint groups.

Focusing on the differences between codecs, when employing the same flooding configuration, the H.264 compressed video has a lower packet arrival rate than the H.265 one due to the higher bitrate that H.264 injects into the network.

The best flooding configuration is different for each codec as they have different traffic demands. The best configuration for the H.264 codec is the Distance-Based algorithm with a configuration of $C = 1$, $MinDistance = 50m$, and $MaxTime = 200ms$. However, the H.265 codec has a lower requirement in terms of bandwidth, and it allows increasing the value of C to add more redundancy without increasing the number of collisions. Figure 6.5 shows that a Distance-Based flooding algorithm with a configuration of $C = 2$, $MinDistance = 50m$, and $MaxTime = 333ms$ is able to achieve the best packet arrival rate in medium and low-density scenarios while maintaining a good performance in the high-density scenario.

6.5.3 Error resilience and impact on the video quality

As in the previous section, we have selected the optimum flooding configuration for each codec to show the frame loss as well as PSNR results.

Figure 6.6 shows the frame loss when employing the best configuration for each codec. While the H.265 codec is able to maintain frame loss at low levels, the H.264 codec suffers an extremely high frame loss when packet losses occur. In fact, the H.264 decoder is often unable to decode several consecutive frames. Although the packet loss difference for both codecs is not very high, H.264 suffers from packet losses in almost every frame, being the decoder unable to properly reconstruct the original video despite using er-

6.5 Results

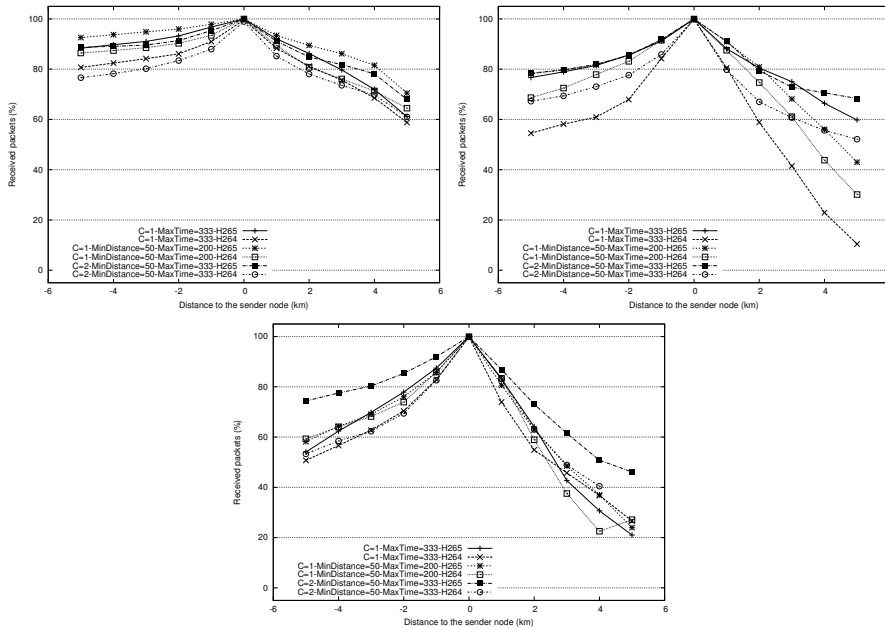


Figure 6.5: Packet delivery ratio. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).

ror recovery techniques like frame-freezing. On the other hand, the H.265 decoder is able to compensate for packet losses with the frame-freezing technique, being able to provide a high number of decoded frames, even though they may have some glitches that affect the PSNR and the visual quality.

When the vehicle arrival rate equals to 1s (high-density scenario) the H.264 codec is able to decode all the frames of the video when flooding backwards, while in the same scenario, when flooding forward (from kilometer 5 to 10), significant frame losses occur even for similar packet loss values. To gain further insight into this phenomenon, we have analyzed the packets loss patterns, detecting differences in terms of the frame types affected. In particular,

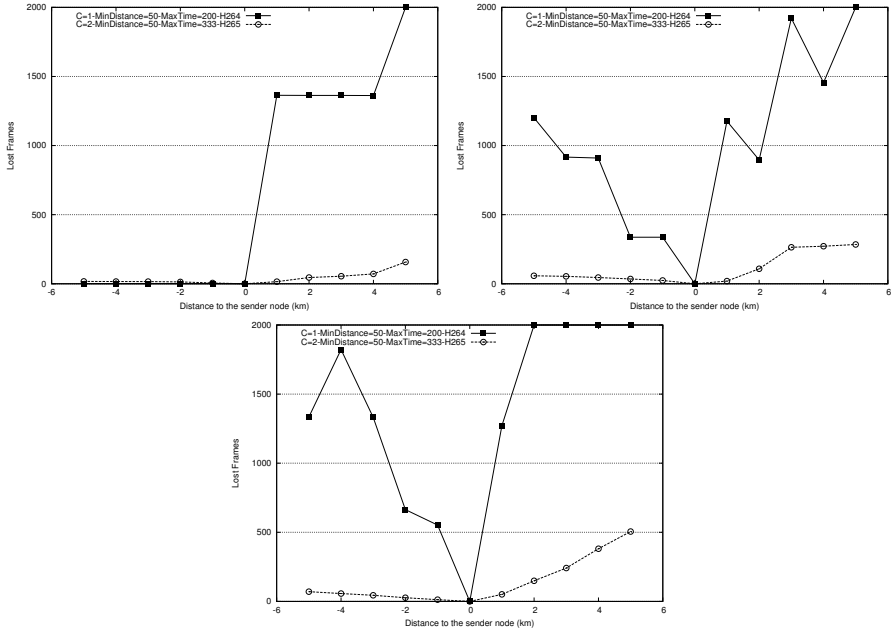


Figure 6.6: Frame loss. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).

more I-frame losses occur in the forward transmission, being the I-frames type crucial to achieve a proper decoding of the video, while the P-frames and B-frames (frames that encode differences between the current frame and the previous I-frames) are not so critical.

Figure 6.7 shows the PSNR for the decoded frames. Although in some cases H.264 performs better than H.265, we have to take into account the differences in the number of decoded frames, since H.264 is only able to decode frames with a reduced packet loss these frames have a good quality, while H.265 is usually able to decode the entire video sequence, and so, the decoded frames have glitches that affect the PSNR. This explains why the H.264 decoded frames

6.5 Results

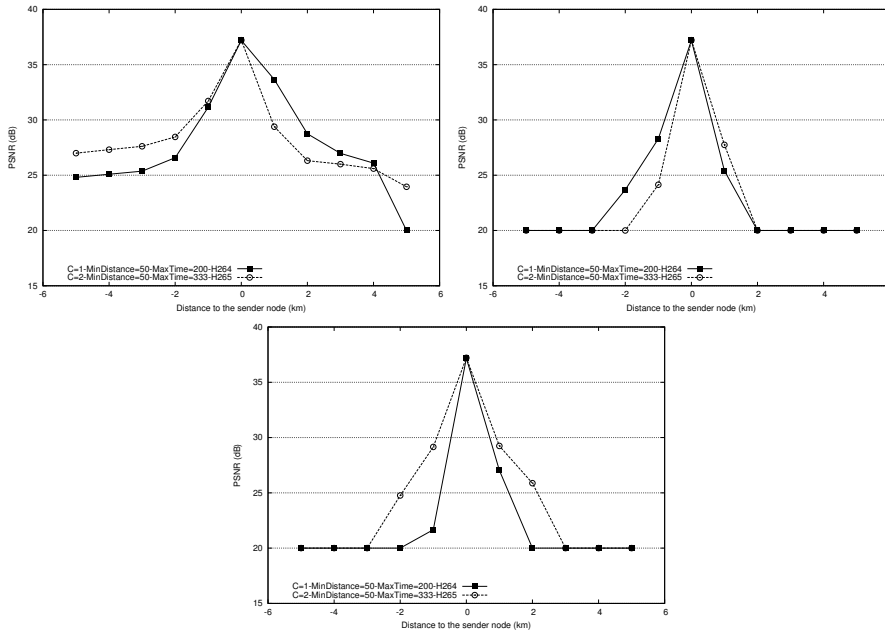


Figure 6.7: PSNR. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).

have better quality than the H.265 decoded ones, on average.

When both decoders have similar frame loss ratios, H.265 performs slightly better than H.264 in terms of PSNR.

These graphs highlight that H.265 outperforms H.264 in terms of both frame loss and PSNR quality. From the user perspective, the video flooded using the H.265 codec is able to provide a much smoother experience than H.264 encoded videos.

For informative purposes, some samples of the decoded videos are made available for download¹.

¹<http://www.grc.upv.es/WCNC-2014/videos.html>

6.6 Conclusions and future work

In this paper we evaluated the user-perceived quality of real-time V2V video flooding in highway scenarios providing frame loss and PSNR data for the widely used H.264 codec and for the new H.265 video codec.

In our experiments we compared two flooding algorithms and selected the best configurations for each of the codecs in terms of packet loss. It is noticeable that, due to the high level of medium congestion produced by the video stream, the better flooding algorithms are not the most aggressive ones, but those that make a more clever use of the medium.

After selecting the best configurations, a detailed analysis in terms of frame loss and PSNR for both of the codecs and for both highway directions is provided.

H.265 has shown to perform better than the H.264 codec, being more robust under high packet loss levels. Additionally, H.265 is able to maintain a controlled frame loss in all the situations, while H.264 is unable to provide an adequate frame rate to achieve a good user experience. In terms of PSNR for the decoded frames, both compression algorithms are unable to sustain the PSNR over $30dB$ for distances greater than 1km, but the H.265 codec is able to provide a better visual quality than the H.264 for a comparable frame loss rate.

Reducing the packet loss in general, especially losses associated with key-frames to avoid massive frame loss, is a key factor to achieve a good user experience. So, as a future work, we plan to design a new flooding algorithm that takes all these factors into account.

Publication F: V2X Solutions for Real-time Video Collection

Alvaro Torres, Yusheng Ji, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *V2X Solutions for Real-time Video Collection*. 11th Annual Conference on Wireless On-demand Network Systems and Services (WONS). Pages 116 to 121. April 2014

Abstract

Quickly identifying the severity of highway accidents, as well as the resources required to assist the people involved in those accidents, is a basic requirement for future intelligent transportation systems. In this context, vehicular communication technologies currently being standardized are able to provide novel solutions to address this problem.

In this work we study the feasibility of combining vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications to deliver a video stream from the accident to the traffic authorities. Our approach relies on vehicles as data relays, thus having the additional advantage of providing drivers with a clear view about the accident, thereby helping to reduce stress and improving traffic flow.

An experimental analysis comparing different traffic flooding mechanisms for wireless networks show that the proposed system is viable for highways with moderate/high amounts of traffic, although highlighting the need for more efficient mechanisms specifically addressing broadcast propagation in highway environments.

7.1 Introduction

The research community has been interested in Vehicular Ad-Hoc Networks (VANETs) for several years since the deployment of this type of networks will be able to provide significant improvements in terms of road safety, as well as to obtain valuable real-time traffic information [44]. VANETs include both vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications, and the combination of both is often referred to as V2X.

Several types of applications and protocols have been proposed to propel the possibilities that this type of network provides [55]. However, every proposed application and protocol has to cope with the problems and characteristics inherent to this type of networks, such as high relative speeds, Doppler effect, low transmission rate, etc.

Automatic accident warning and notification has become a critical safety application in VANETs [35], where the majority of protocols

adopted flooding techniques to warn all the nodes, as well as the traffic authorities, about the accident.

Other type of applications useful for everyday life relies on DTN protocols to spread traffic information, such as traffic jams and pollution levels in different areas of the city [51], [36]. Both applications have in common that the amount of information to transmit can be considered “low”, and both require that the information is able to reach its final destination (high reliability).

Additionally, several studies have built upon the idea of providing Video On Demand (VoD) to vehicle users with a combination of V2V and V2I. Vehicles are only acting as receivers and, although video delivery implies a high amount of information, this can be scheduled and distributed among a set of RSUs, and stored in the vehicle’s buffer.

The recent approval of the new H.265 video compression standard [2], which intends to replace the widely used and well-known H.264 standard [1], provides a new opportunity for real-time video transmission in critical contexts. The new standard, which outperforms the old one achieving the same video quality with only 50% of the bit-rate, is expected to become an enabling technology when attempting to provide real-time video transmission in vehicular networks; thus, H.265 was selected to carry out the different experiments in the scope of our work.

The motivation of this paper is to address the challenges that arise when attempting to provide an innovative service in vehicular environments: vehicles involved in traffic accidents shall produce a high amount of information (a video sequence), which is of interest to all the nodes in the network (both vehicles and RSUs), and this information flow must experience a low delay (soft real-time).

This work does not attempt to present an application that makes use of this type of traffic, focusing instead on evaluating the effec-

tiveness of different flooding schemes with the purpose of achieving a real-time video transmission over multiple hops under different vehicle densities. Based on the results achieved, we will assess the feasibility of distributing a live video stream between vehicles at long distances using flooding. In our study we take into consideration both the packet delivery ratio and the end-to-end delay experienced by video data.

The rest of the paper is structured as follows. In section 7.2 we review the state of the art in terms of both flooding in wireless networks and video transmission over VANETs. Afterwards, in section 7.3, the different flooding schemes used for testing are described. In section 7.4 we provide details about the simulation scenario, as well as an overview of the methodology adopted. Section 7.5 presents the obtained results and, finally, section 7.6 presents the conclusions obtained and discusses future works.

7.2 Related work

The IEEE 802.11p protocol is the standard for Vehicle-to-Vehicle (V2V) communications, and it is based upon the IEEE 802.11 protocol and its quality of service extensions, IEEE 802.11e. So, IEEE 802.11p provides a contention-based broadcast mechanism, and its behavior is closely related to IEEE 802.11; therefore different solutions proposed for the original standard can be adapted to this new type of networks.

During the past few years, several authors have proposed different algorithms to achieve efficient flooding in MANET and VANET environments, proposing several ways of controlling the broadcast storm problem.

Yu-Chee Tseng et al. [52] proposed and improved versions of the Counter-Based, Distance-Based, and Location-Based schemes.

These schemes have some weak points, such as failing to provide any kind of delivery guarantee. Nevertheless, they have some interesting features including (i) low overhead, (ii) being highly adaptable to different conditions, and (iii) providing a completely autonomous broadcast system.

Martínez et al. [34], following the guidelines of the Distance-Based scheme, proposed a flooding mechanism that takes into account the specificities of VANETs; in particular, they tweak it to provide a fast dissemination of accident alerts in urban scenarios by using information such as the street layout to achieve a smarter flooding. For highway scenarios their algorithm does not significantly differ from the Distance-Based approach. Also, although a really high warning rate is achieved, they do not guarantee the alert delivery to every node in the network.

Other authors, like Osafune et al. [40], introduce the concept of “Backfire” for flooding schemes. This concept proposes that, when a node receives a copy of a message from another node which is considered to be a better retransmitter, the first node cancels the retransmission of the packet, thereby achieving reductions in terms of sent messages and channel contention.

Ros et al. [43] try to ensure the message delivery by addressing the problem of temporary disconnections which occur in VANET scenarios. They make use of the beaconing system to piggyback acknowledgements in beacon messages, allowing nodes to start the retransmission of a message when some neighbor has not received the alert message.

With the same idea of making use of the beaconing system to ensure a proper message delivery through acknowledgment piggybacking, Na Nakorn et al. [37] present DECA and DECA-Bewa. Both flooding schemes requires a modified beaconing system, being the later characterized by a variable beacon interval time. Additionally, the flooding algorithm is only based on 1-hop neighbor

density information, thereby avoiding the use of GPS. Authors claim that GPS precision and availability is critical for GPS-based flooding schemes.

F. Soldo et al. [47] presented the SUV protocol, a distributed solution to disseminate video streams in VANETs. The protocol proposes dividing the neighbors into four sectors, and selecting as a candidate for rebroadcasting one node in every sector, although a special MAC layer is required to support TDMA scheduling; such requirement prevents its implementation on actual IEEE 802.11p devices.

In this work we will compare proposals from different authors in order to determine which is more effective when attempting to deliver high amounts of data (in our case video data) through flooding. The proposals chosen for our experiments are described in more detail in the next section.

7.3 Description of the flooding schemes

In this section we will briefly describe the behavior of the different flooding algorithms we have implemented in our simulation platform for testing. We have split the algorithms in two groups: basic schemes and adapted schemes, being the latter a slightly more sophisticated group of schemes compared to the former.

7.3.1 Basic schemes

7.3.1.1 Counter Based Scheme

This scheme was one of the first solutions proposed to effectively reduce the broadcast storm problem. The main idea is that every

node resends a packet until it has received C copies of that packet from other nodes.

To avoid a large number of collisions when receiving a packet, every node waits from a random time before resending it, so it can listen to the medium and wait for the C counter to increase.

All the flooding schemes presented in this paper, except DECA, adopt the concept of receiving C copies of a packet to stop re-broadcasting it.

This scheme has several advantages over other solutions. The first one is that it is quite easy to implement, not having any relevant requirements except for a wireless card. The second advantage is the possibility of tuning the algorithm by modifying both the maximum amount of time a node can wait to rebroadcast a packet, and the number of copies a node should hear to stop rebroadcasting; this way, merely by modifying a simple parameter, we are able to increase the redundancy. The third advantage is that it is a good reference for other algorithms and, in case of problems with any of the required modules (GPS or Beaconing system), it can be used as a fall-back scheme without dramatically degrading the flooding performance.

The reason for selecting the Counter-Based scheme instead of other sophisticated schemes such as [54], [43] and [37], is the type of traffic we intend to transmit (video traffic), especially when fast data spreading is more relevant than high reliability.

7.3.1.2 Distance Based Scheme

This scheme builds upon the Counter Based Scheme by adding a new requirement: a positioning system (GPS).

Due to this new requirement, we can easily improve the intelligence

of the algorithm to decide which node can be a better next-hop when rebroadcasting a packet.

Since the algorithm should remain autonomous, and since the decision about whether to rebroadcast a message should be taken independently by each node, this scheme makes use of the waiting time to resend messages. The time a node waits for before resending is inversely proportional to the minimum distance between the original node and the resending node. This way, nodes are virtually ordered according to the additional area of coverage, prioritizing those nodes whose additional area is supposed to be bigger.

To implement this scheme we need some kind of location mechanism, such as GPS, and the packets should be marked with the original sender node position and the resending node position.

This mechanism allows a rapid spreading of the packets while obtaining a profitable message forwarding process in terms of extra coverage area achieved by every new transmission.

7.3.2 Adapted schemes

7.3.2.1 DECA

DECA [37] is a flooding scheme that does not rely on positioning mechanisms to spread information. Instead, it makes use of the beaconing system to estimate the 1-hop neighbor density and, thus, select as resending node the neighbor with more 1-hop neighbors. Also, it makes use of the beaconing mechanism to piggyback acknowledgement messages, thereby reducing the channel resources consumed.

DECA was adapted to video streaming particularities by including in the beacons information about the last packet properly received.

7.3.2.2 Backfire Scheme

The basic behavior of this algorithm is similar to the Distance Based Scheme. As proposed in [40], the major improvement introduced has to do with an analysis of area covered by neighboring nodes. In particular, if a node receives a packet from another node that is supposed to provide more additional coverage area, it stops rebroadcasting that packet. This strategy allows reducing the number of collisions by minimizing the number of unnecessary transmissions.

7.4 Scenario and Methodology.

The simulation environment is composed by three main components:

- OMNeT++ [6], an event-driven simulator which provides a base for implementing several types of models.
- INET framework [3], an implementation of the different network models for the OMNeT++ simulator, which includes several network models, from the physical to the application layer.
- SUMO (Simulation of Urban MObility) [14], which provides realistic vehicle behavior. SUMO runs coupled with the OMNeT++ simulator by using TRaCI, allowing several mobility parameters, such as vehicle speed, to be changed in simulation time.

The transmission range in the INET framework is not defined as a fixed distance. Instead, it requires tuning different parameters

such as the frequency or the level of attenuation with distance. To achieve the highest degree of similarity with reality we adopted the parameters proposed by Báguena et al. [11].

For a more realistic mobility behavior that includes vehicle overtaking, we defined a set including different vehicle types with an associated probability of occurrence. All this data can be seen in table 7.1.

To perform our experiments with predefined vehicle densities we used VACaMobil [12], a tool that allows defining a vehicle arrival rate, maintaining a stable mean number of vehicles throughout the entire simulation.

Table 7.1: Vehicle types and associated probability.

Vehicle Type	Maximum Speed (m/s)	Length (m)	Probability
Truck	25	12	0.10
Car	33	4	0.79
Slow Car	25	5	0.10
Fast Car	39	4	0.01

Additionally, the SUMO step value was configured to $0.1s$ in order to achieve a more realistic scenario. Notice that, when vehicles move at highway speeds ($30m/s$), the default SUMO step ($1s$) provides a level of granularity that is too coarse for our experiments. By setting this step to a lower value, position updates become more frequent, allowing the vehicles' movements in the simulation become smoother, and making communications between vehicles more realistic.

7.4.1 Scenario

Figure 7.1 shows the simulated scenario, which is a 10 kilometer straight highway with two ways and two lanes per way.

A set of RSUs has been deployed at every kilometer, and the distance between the RSUs and the road is $5m$. RSUs do not cooperate in the flooding mechanism, being mere traffic sinks used to measure the expected quality at every kilometeric point of the highway. This strategy allows measuring the expected video quality when deploying RSUs with different spacing between them.

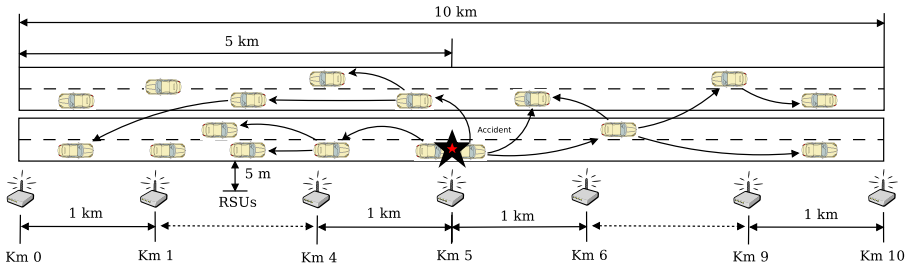


Figure 7.1: Highway scenario.

For our simulations we varied the vehicle arrival rates to compare the effectiveness of the protocols at different vehicle densities.

When the number of vehicles in the network becomes stable we schedule an accident at the centre of the scenario ($5km$ point). Since vehicles driving in the direction of the accident vehicle attempt to move from kilometer 0 to kilometer 10, the accident produces a small traffic jam from kilometers 0 to 5 on both of the lanes. For vehicles moving in the opposite direction, traffic continues to flow undisturbed.

Concerning video traffic generation, the damaged vehicle immediately starts transmitting video through flooding.

7.4.2 Methodology

To evaluate the four different algorithms under the same traffic conditions, we run a set of 15 repetitions per configuration with different node mobility patterns, thereby achieving a good data significance.

We compare the effectiveness of the different flooding schemes in terms of packet arrival ratio and delay.

7.4.2.1 Simulation setup

The different flooding schemes have been tested under three vehicle densities. These densities are achieved by setting an exponential Inter-Arrival Time of vehicles equal to $0.25s$, $1s$, and $3s$ respectively. Such Inter-Arrival Times correspond, approximately, to a mean density of 100, 30, and 10 *vehicles/km*. These three vehicular densities have been labeled as “High”, “Medium”, and “Low” in the figures of this paper.

Additionally, for each algorithm and vehicular density, we tested all the possible combinations for the C and *Waiting Time* parameters. Possible C values are 1, 2, 3, 4, and 5; *Waiting Time* values can be 100, 200, 300, 400, and 500 ms .

All the different results presented in the next section correspond to the best possible configuration of every algorithm, for each density, in terms of parameters C (Number of copies required to stop resending) and *Waiting Time* (Maximum time that a node waits to resend a message). The optimal parameters for the selected schemes are shown in table 7.2.

As expected, higher node densities are associated with lower count values (parameter C), and in general lower timeout values as well, since the existence of more nodes makes the process quicker and more resilient.

Table 7.2: Parameters for best-case simulations.

Density	High		Medium		Low	
	C	Time (ms)	C	Time (ms)	C	Time (ms)
Counter	1	400	2	400	3	500
Distance	1	300	2	500	2	500
Backfire	2	300	3	200	5	300

7.5 Results

In this section we present the results from our experiments, which were obtained using the simulation setup and methodology described above. We focus on two different metrics: packet delivery ratio and end-to-end delay. These metrics are assessed on RSUs deployed every kilometer.

Figure 7.2 (left) shows the percentage of received packets for the different flooding schemes when vehicle density is high. We can appreciate how almost every single flooding scheme is able to achieve more than 92% of packet arrival ratio, even at a distance of 5km from the accident (kilometers 0 and 10).

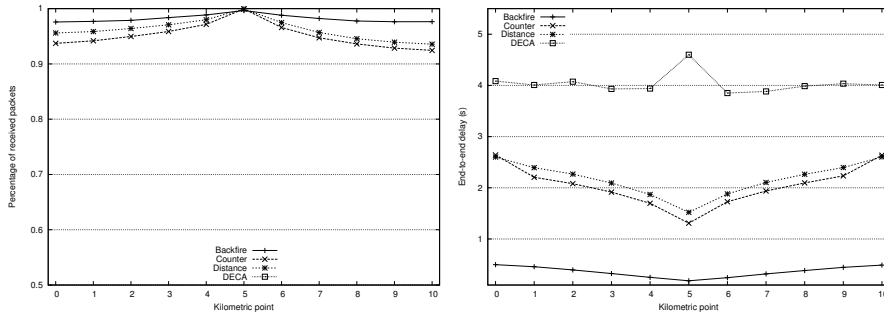


Figure 7.2: High density scenario. Percentage of packets received.

We can see that the Backfire scheme performs better than any

other scheme due to the reduced number of collisions produced. DECA is the worst performing solution among all presented algorithms, with packet delivery ratios ranging from 21.2% to 48.4% (values below the range shown in the figure). Due to the high requirements of video traffic in terms of bandwidth, a significant amount of collisions occurs when vehicles piggyback the information about missing packets. Notice that, if some nodes experience packet losses, a high number of retransmissions are scheduled by different nodes, thus increasing the chances of collapsing the wireless medium.

In terms of delay, if we look at the right part of figure 7.2, the differences between the selected algorithms are very noticeable. The Backfire scheme is able to deliver the packets in a very short time (less than 500 ms), while for Counter and Distance schemes the delay is quite higher (about 2.7 s). As a high number of collisions occur, the wireless medium is extremely congested, and so the delay experienced by DECA is quite high; in fact, DECA's delay values are about eight times higher than the average delay achieved when using the Backfire scheme.

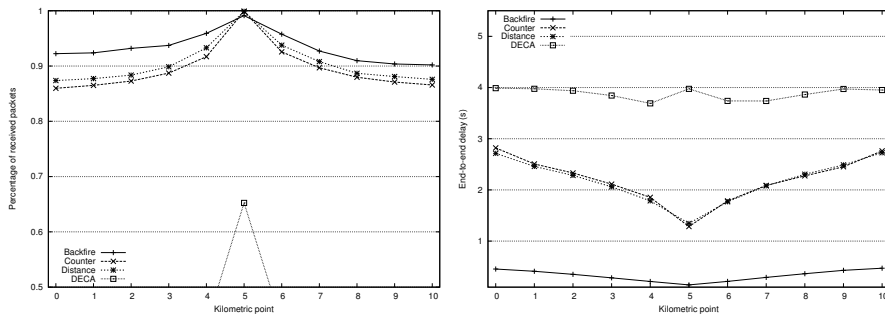


Figure 7.3: Medium density scenario. Percentage of packets received.

Figure 7.3 shows the results for the medium density scenario. In

7.5 Results

this scenario, the Backfire flooding scheme is able to provide the best results up to $4km$, achieving a delivery ratio of more than 90%. If we focus on the Counter and Distance schemes, we can see that the delivery ratio drops below 90% at a distance of $3km$. Notice that packet losses greater than 10% are prone to cause frequent video artifacts or even severe decoding problems.

Concerning delay, the values achieved do not significantly differ from those obtained under high node densities.

Again, we find that the behavior of the DECA-based scheme is still highly compromised, although slight improvements are detected.

Finally, figure 7.4 shows the results for the low density scenario. We find that now the Distance-based scheme achieves better results than the Backfire scheme, although all the flooding schemes evaluated are unable to achieve an acceptable percentage of received packets from the video decoding perspective (more than 90%) when the density of vehicles is low.

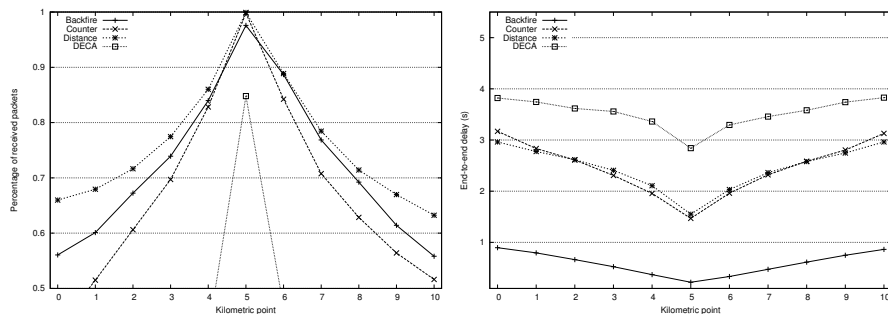


Figure 7.4: Low density scenario. Percentage of packets received.

If we focus in the delay for the received packets, we can see that the times are maintained with respect to the high and medium density scenarios, except for the DECA scheme which experiences a slight improvement.

Summarizing, we find that flooding schemes like DECA, which make use of the beaconing system to request missing packets, are unsuitable for real-time video flooding due to the packet losses and high delays introduced. We also find that schemes like Backfire, which aggressively try to reduce collisions in the medium while introducing low delays, are more suitable for real time video transmission, providing better overall results than the other schemes tested in terms of both packet delivery ratio and delay.

7.6 Conclusions and future work

Future intelligent transportation systems are expected to provide sophisticated applications to improve traffic safety and make driving a more effort-free and amenable task. To support these services, solutions combining V2V with V2I communications can help to alleviate the infrastructure deployment cost. Such costs are especially significant in highway environments, where several kilometers of road should be served while avoiding very high costs.

When accidents occur in a highway, providing the traffic authorities a real-time video from the accident premises can be quite helpful to assess accident severity and deploy the adequate services (fireman, ambulances, traffic regulation personnel, etc.). However, performing real-time video streaming in vehicular environments while relying on flooding for data propagation is a quite complex task, especially when considering the low channel capacity available and the high number of nodes involved.

In this paper we analyzed the viability of delivering video towards RSUs deployed at different distances from the accident. In our study we used four different flooding schemes, and provide both packet arrival ratio and end-to-end delay results under low, medium and high traffic congestion levels. Experimental results

show that the Backfire algorithm, or even simpler solutions such as the distance-based algorithm, are able to provide acceptable performance results. Specifically, our findings show that:

- Under high vehicle densities, video delivery to RSUs located up to 5km away from the accident position is feasible, being the packet loss rate maintained below 10% using all available protocols (except DECA).
- Under medium vehicle densities, video delivery to RSUs located up to 5 km away is possible only when using the Backfire scheme, being that simpler schemes, like counter-based and distance-based, are only able to provide an acceptable delivery rate up to 2 km away.
- Under low vehicle densities, an effective video delivery is not achievable with any of the schemes tested, being quality significantly degraded just 1km away from the accident location.

As future work we plan to develop and evaluate new flooding schemes optimized for video delivery in highway scenarios, especially under medium and low vehicle densities.

**Publication G:
Evaluation of Flooding Schemes for
Real-time Video Transmission in
VANETs**

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano, Pietro Manzoni and Yusheng Ji. *Evaluation of Flooding Schemes for Real-time Video Transmission in VANETs*. Ad Hoc Networks, Volume 24. Pages 3-20, January 2015

Abstract

Long-distance real-time video flooding over Vehicular Ad-hoc Networks (VANETs) is challenging due to the dynamic nature of vehicular networks. Real-time video transmission has high requirements in terms of bandwidth and delay, while VANETs are characterized by very limited radio resources and high mobility.

Our objective is to compare ten different flooding schemes specially designed for this task and select the best in terms of packet arrival ratio and Peak signal-to-noise ratio (PSNR). Additionally, we propose Abstract Copies Distance Based (ACDB), an improved flooding scheme, to cope with variable vehicle density situations.

Furthermore, to ensure a good behavior under all types of circumstances, we also study the impact of Global Positioning System (GPS) drift on these schemes. Simulation results show that long-distance real-time video flooding transmission is feasible under certain vehicle density conditions and the quality achieved is enough to allow using such solutions in future applications.

8.1 Introduction

The research community has been interested in Vehicular Ad-Hoc Networks (VANETs) for several years since the deployment of this type of networks will be able improve road safety significantly [24], as well as obtain valuable real-time traffic information [44].

Several types of applications and protocols have been proposed to propel the possibilities that this type of networks provide [55], [25]. However, every proposed application and protocol has to cope with the problems and characteristics inherent to VANET environments, such as high relative speeds, the Doppler effect, low transmission rates, etc.

Automatic accident warning and notification has become a critical safety application in VANETs [35], where the majority of protocols adopted flooding techniques to warn all nodes, as well as the traffic authorities, about an accident.

Other types of applications useful in our everyday lives rely on Delay Tolerant Network (DTN) protocols to spread information

about traffic, such as traffic jams, and pollution levels detected in different areas of the city [51], [36]. In both applications, the amount of information to transmit can be considered “low”, and information about how to reach the final destination is required (high reliability).

Additionally, several studies have built upon the idea of providing Video on Demand (VoD) to vehicle users with a combination of Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I). Vehicles only act as receivers, and although video delivery implies a high amount of information, the video can be scheduled and distributed among a set of Road Side Units (RSUs) and stored in the vehicle’s buffer.

The recent approval of the new H.265 video compression standard [2], which is intended to replace the widely used and well-known H.264 standard [1], provides a new opportunity for real-time video transmission in critical contexts. The new standard, which outperforms the old one by achieving the same video quality with only 50% of the bit-rate [31], is expected to become an enabling technology when attempting to provide real-time video transmission in vehicular networks; thus, H.265 was selected to carry out the various experiments in the scope of our work.

The motivation of this paper is to address the challenges that arise when attempting to provide an innovative service in vehicular environments: vehicles involved in traffic accidents produce a high amount of information (a video sequence), which is of interest to all the nodes in the network (both vehicles and RSUs), and this information flow must reach its destination within a low delay (soft real-time).

This work does not attempt to present an application that uses this type of traffic but, instead focuses on evaluating the effectiveness of different flooding schemes in order to achieve a real-time video

transmission. Thus our objective is to study the possibility of achieving live video streaming between vehicles over multiple hops under different circumstances, such as different vehicle densities and different degrees of GPS accuracy, while having RSUs deployed along the road as the final destination for data. With this purpose in mind, we implemented and compared some existing flooding schemes and proposed new ones.

The rest of the paper is structured as follows. Section 8.2 reviews the state of the art in terms of both flooding in wireless networks and video transmission over VANETs. Afterwards, Section 8.3 describes all the different flooding schemes. Section 8.4 provides the details of the simulation scenarios as well as an overview of the methodology adopted. Section 8.5 presents the obtained results in both highway and urban scenarios, and finally, Section 8.6, summarizes the conclusions we obtained.

8.2 Related work

IEEE 802.11p protocol is the standard for Vehicle to Vehicle (V2V) communications, and it is based upon IEEE 802.11 protocol and its quality of service extension, IEEE 802.11e. Thus, IEEE 802.11p provides a contention-based broadcast mechanism, and its behavior is closely related to IEEE 802.11; therefore, different solutions proposed for the original standard can be adapted to this new type of network.

During the past few years, several authors have proposed different algorithms to achieve efficient flooding in MANET and VANET environments, proposing several ways to control the broadcast storm problem.

Yu-Chee Tseng et al. [39] presented some basic algorithms to solve this problem. The same authors published an improved ver-

sion of the same algorithms by adding adaptive conditions to further reduce the broadcast storm problem [52]. In particular, they proposed and improved versions of the Counter-Based, Distance-Based, and Location-Based Schemes. These schemes have some weak points, such as failing to provide any kind of delivery guarantee. However, they have some interesting features including (i) having low overhead, (ii) being highly adaptable to several conditions, and (iii) providing a completely autonomous broadcast system.

Martínez et al. [34], following the guidelines of the Distance-Based Scheme, proposed a flooding mechanism that takes into account the specificities of VANETs. Specifically, they tweaked it to disseminate accident alerts quickly in urban scenarios by using information such as town layout to achieve a smarter flooding. For highway scenarios, the algorithm does not significantly differ from the Distance-Based Scheme. Although a really high warning rate is achieved, it does not guarantee the alert delivery to every node in the network.

Other authors, like Osafune et al. [40], introduced the concept of “Backfire” for flooding schemes. This concept proposes that, when a node receives a copy of a message from another node that is considered to be a better re-transmitter, the first node cancels the retransmission of the packet, thereby reducing sent messages and channel contention.

To increase both efficiency and the message delivery ratio, Wu et al. [54] presented a sender-oriented flooding scheme called FUZZBR, which proposes a mechanism of next-hop selection based on fuzzy logic. To make an adequate node selection, they gather information by using a beaconing system, and then classify the neighbors in accordance with the distance, degree of mobility, and RSSI of the received messages. FUZZBR also tries to guarantee message delivery using an ack-based timeout mechanism, i.e. if a selected

node does not rebroadcast the message, the source node will retry the sending.

Ros et al. [43] tried to ensure the message delivery by addressing the problem of temporary disconnections that occur in VANET scenarios. They use the beaconing system to piggyback acknowledgements in beacon messages, allowing nodes to start the retransmission of a message when some neighbors have not received the alert message.

On the basis of the same idea of using the beaconing system to ensure a proper message delivery through acknowledgment piggybacking, Na Nakorn et al. [37] present DECA and DECA-Bewa. Both flooding schemes require a modified beaconing system, being the later characterized by a variable beacon interval time. Additionally, both flooding schemes are solely based on 1-hop neighbour density information, thereby avoiding the use of GPS. They claim that GPS precision and availability are critical for GPS-based flooding schemes.

To avoid the GPS availability problem which typically occurs in tunnels, Yan et al. [55] proposed a grid-based on-road localization system. According to their results, the positioning error can be higher than 10 *m*. Additionally, they update the node position every two seconds, and during that interval, the node position can be considered “old”. In fact, even vehicles with GPS receivers take some time to update their position, which can lead to high deviations from the real position. Thus, the performance of GPS-based flooding algorithms has to be evaluated under these circumstances.

Although many recent articles have addressed video transmission over VANETs, most authors propose video transmission as an entertainment technology alone, within the scope of “infotainment”. In these situations the video is streamed from Road Side Units (RSU) to vehicles. Cuomo et al. [20] propose the creation of a vehicular

Backbone Network (VBN) for high throughput content distribution on highways. In contrast, this work focuses on distributing live video streaming between vehicles and from vehicles to some distant RSU. Meng Guo et al. [25] presented several scenarios where live video streaming between vehicles is both feasible and desirable.

Soldo et al. [47] presented the Streaming Urban Video (SUV) protocol, a distributed solution to disseminate video streams in VANETs. The protocol proposes dividing the neighbors into four sectors and selecting one node in every sector as a candidate for re-broadcasting; although a special MAC layer is required to support TDMA scheduling. This requirement prevents its implementation on actual IEEE 802.11p devices. Additionally, it requires very precise clock synchronization, which should be achieved using GPS. Nevertheless, as mentioned before, we assume that nodes can lose GPS signals, which are prone to cause clock synchronization problems.

Overall, although the aforementioned studies provide a good number of statistics, none presents actual video quality results such as PSNR, because they do not take into account decoding problems that typically arise, such as the interdependence between frames.

Piñol et al. [41] proposed the first approach towards a proper simulation environment capable of representing real-time video transmission in VANETs. Authors present a simulation platform and PSNR results, following the guidelines presented by Seeling and Reisslein [45], thereby providing a useful approach on how to simulate video transmission accurately.

With respect to previous works, in this paper we evaluate the performance of different real-time video flooding schemes having a vehicle as a source of the video, evaluating the error resilience in terms of PSNR for the new H.265 video coding standard, as well as the required conditions, in terms of distance and vehicular density

for obtaining a good video quality. Also, we propose an adaptive flooding algorithm, and we assess the robustness of the different flooding schemes when the positioning system is not precise.

8.3 Description of the flooding schemes

In this section, we will briefly describe the behavior of the different flooding algorithms implemented. Figure 8.1 shows the different requirements of each scheme in terms of GPS or a beaconing system, as well as the original scheme upon which each one is based.

As shown, all the different algorithms require GPS, except the DECA-Based and Counter-Based ones, whereas the beaconing system is required by every algorithm except the Counter-Based, the Distance-Counter-Based, and Backfire Schemes.

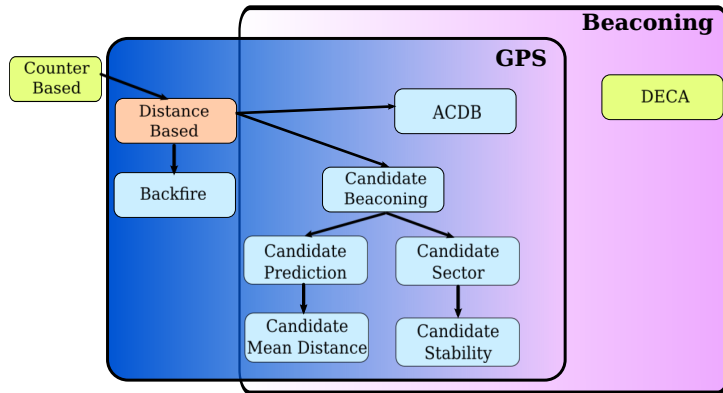


Figure 8.1: Inheritance diagram of the different flooding schemes.

8.3.1 Basic schemes

8.3.1.1 Counter Based Scheme

This scheme was one of the first proposed to reduce the broadcast storm problem effectively [52]. The main idea is that every node resends a packet until it has received C copies of that packet from other nodes.

To avoid a large number of collisions when receiving a packet, every node waits for a random time (*Waiting Time*) before resending, so it can listen to the medium and wait for the C counter to increase.

All the flooding schemes presented in this paper, except DECA, will adopt the main idea of receiving C copies of a packet to stop rebroadcasting it.

This scheme has several advantages over other solutions. The first is that it is quite easy to implement, not having any other relevant requirements except for a wireless card. The second advantage is the possibility of tuning the algorithm by modifying both the maximum amount of time a node can wait to rebroadcast a packet and the number of copies a node should hear to stop rebroadcasting; this way, merely by modifying a simple parameter, we are able to increase the redundancy. The third advantage is that it is a good reference for other algorithms and, in the case of problems, with any of the required modules (GPS or a beacons system), it can be used as a fall-back scheme without dramatically degrading the flooding performance.

The reason for selecting the Counter-Based Scheme instead of other more sophisticated schemes, such as [54], [43] and [37], is the type of traffic we intend to transmit (video traffic), especially when fast spreading is more relevant than high reliability.

8.3.1.2 Distance Based Scheme

Based on the Counter Based Scheme, this scheme [52] adds a new requirement: a positioning system (i.e. GPS).

Due to this new requirement, we can easily improve the intelligence of the algorithm to decide which node can be a better next-hop when rebroadcasting a packet.

Since the algorithm should remain autonomous, and since the decision about whether to rebroadcast a message should be taken independently by each node, this scheme uses the *Waiting time* to resend messages. The time a node waits before resending is inversely proportional to the minimum distance between the original node and the resending node. This way, nodes are virtually ordered in accordance with the additional area of coverage, prioritizing those nodes whose additional area is supposed to be bigger.

To implement this scheme, we need some kind of location mechanism, such as GPS, and the packets should be marked with the original sender node position and the resending node position.

This mechanism allows a rapid spreading of the packets while obtaining a profitable message forwarding process in terms of extra coverage area achieved by every new transmission.

8.3.2 Adapted Schemes

8.3.2.1 DECA

DECA (DEnsity-aware reliable broadCAsting protocol) [37] is a flooding scheme that does not rely on positioning mechanisms to spread information. Instead, it uses the beaconing system to estimate the 1-hop neighbor density, and thus select the neighbor with more 1-hop neighbors as the resending node. It also uses the

beaconing mechanism to piggyback acknowledgement messages. DECA was adapted to video streaming requirements by including in beacons information about the last properly received packet. With this information, nodes can trigger the resending of the remaining packets not received by a certain node.

8.3.2.2 Backfire Scheme

This algorithm shares a common basis with the Distance Based Scheme, but now, as proposed by Osafune et al. [40], if it receives a packet from a node that is supposed to provide more additional coverage area, it stops rebroadcasting that packet. This strategy allows a reduction in the number of collisions by minimizing the number of transmissions.

8.3.3 Proposed Schemes

8.3.3.1 Candidate Beaconing

If we use another resource (beaconing system), we can obtain fairly valuable information, such as the position of 1-hop neighbors.

As stated previously, the furthest neighbor is the one that is going to provide more additional coverage area, making packet rebroadcasting more beneficial.

Source-based flooding schemes usually propose a node to rebroadcast the packet. They wait for a certain amount of time to hear the rebroadcast (which serves also as an ACK mechanism), and, in the case of failure, the sender node resends the message. The main problem of these schemes is that they introduce many waiting periods, which makes them unfeasible for real-time video transmission.

Instead, we introduce the concept of “Candidate”. Every time a vehicle sends a packet it selects one or more of its neighbors and proposes them as candidates. These candidates should rebroadcast the packets immediately, while the rest of the vehicles should wait for the minimum amount of time before rebroadcasting the packet. The main difference from pure source-based algorithms is that, in this case, enforcing a minimum waiting time allows the receiver node to face a less congested medium. Thus, after this minimum waiting time, they behave similarly to the Distance Based Scheme. In this proposed scheme, and as a first approach, the candidate is selected simply by selecting the furthest node in our neighbor list.

8.3.3.2 Candidate Prediction

The beaconing system does not usually have an updated information about every neighbor position since position information is only updated when a new beacon is received, which occurs once every few seconds, and so it can easily lead to position deviations higher than 50m.

If every vehicle adds information about its speed and direction to beacons, we can easily estimate the current position of each of our neighbors, and thereby, discard the ones that are further away than the furthest beacon received.

This strategy enables the selection of a better candidate than merely using old positioning information.

8.3.3.3 Candidate Mean Distance

According to the Nakagami Fading Channel model and standard signal propagation theory, the further the node, the more difficult for that node to receive a packet properly.

To improve the candidate selection, and on the basis of the predicted positioning information, the Candidate Mean Distance Scheme selects a candidate that has a high probability of receiving the packet, but also that is far enough to guarantee that the additional area coverage is sufficiently large. Thus, in this case, it selects the closest neighbor for a distance of at least 60% of the distance to the furthest beacon received.

8.3.3.4 Candidate Sector

To improve the critical problem of selecting the best candidate, if the positioning system can provide compass information, the list of neighbors can be divided into several sectors in accordance with their relative position.

As an example, in a highway scenario, a set of two sectors can be defined (front and back). If a packet is received from a node in its “front” sector, it should select the candidate from the “back” sector to improve the coverage area.

This mechanism can be easily extended to cities by adding more sectors, such as: front, back, left, and right. Thus, if a packet is received from the “right” sector, we can select up to three candidates, one for each of the other sectors.

This space division attempts to avoid the spreading of messages back to a position close to the sender node. Previous work also introduced this mechanism, but it requires MAC-layer modifications [47].

In addition, since this scheme is based on the Candidate Beaconing Scheme, the candidate for each sector is the furthest node.

8.3.3.5 Candidate Stability

One of the key factors of good candidate selection is the quality of a certain neighbor as a relay node.

The work of Wu et al. [54], relies on individualized RSSI information. In a real world scenario, that information cannot be obtained through a standard UDP socket, instead relying on the quality of the driver implementation of each manufacturer or on external solutions such as adding a second wireless interface in monitor mode.

To achieve a highly compatible and easy to implement solution, we propose measuring the reliability of each of our neighbors by using the beaconing system. Every node should state, in every sent beacon, the estimated time for the next one, allowing the receivers of that beacon to estimate the amount of lost beacons.

The Candidate Stability Scheme proposes the furthest node with the lowest amount of lost beacons as the candidate. In addition, since it is based on the previous Candidate Sector Scheme, it selects the most stable candidate for rebroadcasting in every sector.

8.3.3.6 Automatic Copies Distance Based (ACDB)

The ACDB Scheme uses the density information provided by the beaconing system, as well as the number of queued packets, to dynamically adjust the value of C and the maximum waiting time available to rebroadcast a packet.

With this scheme, we intend to avoid the candidate selection process present in the sender-based schemes, and we try to provide better support for high and low density environments while maintaining, at least, the same behavior as the other schemes.

The behavior of this scheme is to reduce the value of C and increase the *Waiting time* when the density is high, and to increase

C and reduce the *Waiting time* when the density is low. The first approach attempts to reduce the number of rebroadcasts, and thereby the number of collisions, while the second one attempts to increase the redundancy.

The value of C can range from 1 to 5, meaning that the node will stop rebroadcasting after receiving C extra copies of the message. The *Waiting Time* can range from 100 *ms* to 500 *ms*. The values of parameters C and *Waiting Time* are coupled into five levels of density, being the values $C = 1$, *Waiting Time*= 500 *ms*, $C = 2$, *Waiting Time*= 400 *ms*, and so on.

The local density is estimated by using the number of 1-hop neighbors reported by the beaconing system. Every node classifies the current number of neighbors into five levels, each one corresponding to certain values of C and *Waiting Time*, as defined above. The classification is made in a linear way between 1 and the maximum number of 1-hop neighbors previously detected.

8.4 Scenario and Methodology.

The simulation environment is composed of three main components:

- OMNeT++ [6], an event-driven simulator that provides a baseline for implementing several types of models.
- INET framework [3], an implementation of the different network models for the OMNeT++ simulator which includes several models from the physical to the application layer.
- SUMO (Simulation of Urban MObility) [14], which provides realistic vehicle behavior. SUMO runs coupled with the OMNeT++ simulator by using TRaCI, thereby allowing several

mobility parameters, such as vehicle speed, to be changed in simulation time.

The transmission range in the INET framework is not defined as a fixed distance. Instead, it requires tuning different parameters such as the frequency or the level of attenuation with distance. To achieve the highest degree of similarity with real-life behavior we adopted the parameters proposed by Báguena et al. [11]. The exact parameters are shown in table 8.1.

Table 8.1: Transmission parameters.

Parameter	Value
Frequency	5.9 GHz
Atenuation model	Nakagami
alpha	2.4
Nakagami " m "	1
Broadcast Speed	6 Mbit/s
Transmission Power	18 dBm
Antenna gain	5 dB

For a more realistic mobility behavior that includes vehicle overtaking, we defined a set including different vehicle types with an associated probability of occurrence. All these data are listed in Table 8.2.

To perform our experiments with predefined vehicle densities, we used VACaMobil [12], a tool that allows defining a vehicle arrival rate, maintaining a stable mean number of vehicles throughout the entire simulation.

Additionally, the SUMO step value was configured to $0.1s$ in order to achieve a more realistic scenario, such as when vehicles move at highway speeds ($30 m/s$). The default SUMO step ($1 s$) provides a level of granularity that is too coarse for our experiments.

Table 8.2: Vehicle types and associated probability.

Vehicle Type	Maximum Speed (m/s)	Length (m)	Probability
Truck	25	12	0.10
Car	33	4	0.79
Slow Car	25	5	0.10
Fast Car	39	4	0.01

8.4.1 Simulation Scenario

Figure 8.2 shows the first simulated scenario, which is a 10-kilometer straight highway with two ways and two lanes per way.

A set of RSUs has been deployed every kilometer, and the distance between the RSUs and the road is $5m$. RSUs do not cooperate in the flooding mechanism, being mere traffic sinks used to measure the expected quality at every kilometeric point of the highway.

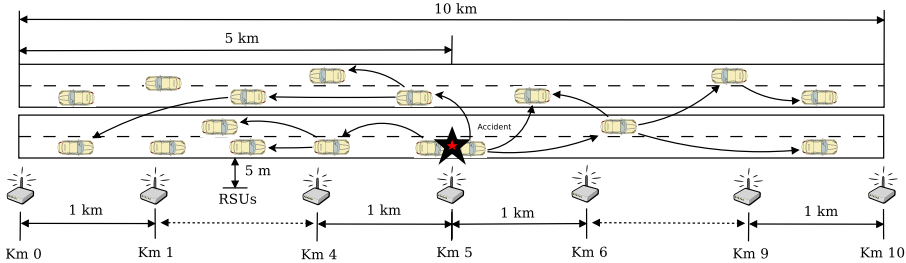


Figure 8.2: Highway scenario.

For our simulations, we varied the vehicle arrival rates to compare the effectiveness of the protocols at different vehicle densities.

When the number of vehicles in the network stabilizes, we schedule an accident at the center of the scenario ($5km$ point). The accident produces a small traffic jam between kilometers 0 and 5. Further-

more, when the scheduled vehicle experiences the accident, it starts the video transmission and disseminates it through flooding.

Figure 8.3 shows the second scenario, which is a 5 square kilometers urban scenario. It corresponds to the downtown of the city of Milan, Italy.

A set of RSUs (blue circles) has been placed in several junctions trying to create five circles around the sender / accident position (red star), in order to obtain an estimation of the quality received at each point of the city.

As in the previous scenario, we varied the number of vehicles while guaranteeing stable densities of 50, 100, and 200 *vehicles/km²*.

8.4.2 Methodology

To evaluate the 10 different algorithms under the same traffic conditions, we run a set of 25 repetitions per configuration with different node mobility patterns, thereby achieving a good data significance.

We first compare the effectiveness of the different flooding schemes in terms of packet arrival ratio and video PSNR. Then, for the highway scenario, we study the impact of the positioning accuracy on the different flooding schemes.

Packet arrival ratio is shown for every scheme, while for PSNR results we selected four representative schemes.

The selected schemes are ACDB, Distance Based, Candidate Stability, and Backfire. The Distance Based Scheme is selected as a reference algorithm. Candidate Stability is selected because it is the best performing scheme among the source-based schemes. Finally, Backfire and ACDB are selected because they are the best performing algorithms in some of the cases.

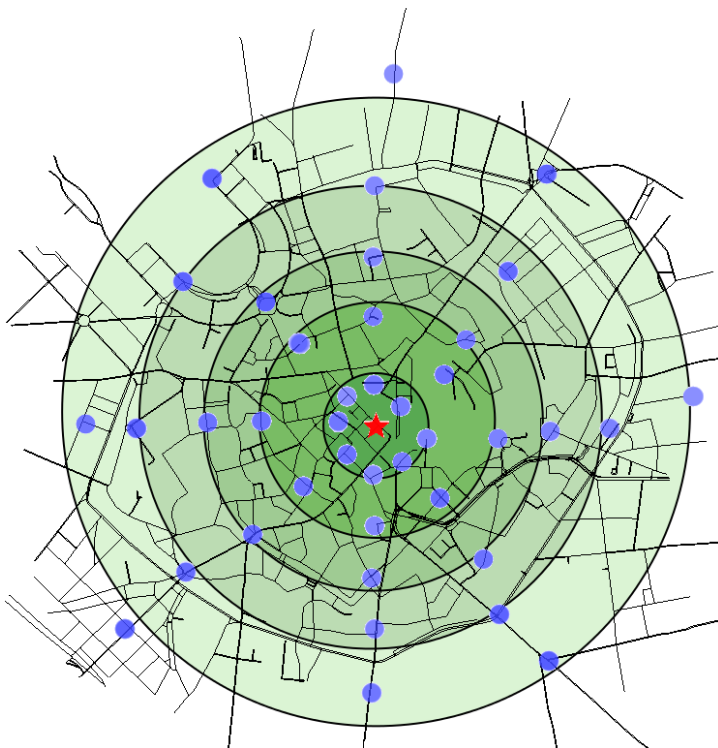


Figure 8.3: Urban scenario.

8.4.2.1 Video Quality Measurement

The actual transmitted video is the CIF version of the “Highway” video sequence, which contains 2000 frames [10]. The video has been encoded using the H.265 reference codec, obtaining a quality level of 37.87 dB. This quality level corresponds to a data rate of 283 kbit/s. The encoder was configured to create RTP packetized streams. To avoid large dependencies between frames, and to try to achieve a better packet loss resilience, key-frames are generated every two seconds.

Additionally, to add real-time constraints, we assume a video buffer of 1s, meaning that a vehicle will start the video playback 1s after the arrival of the first packet and discard every packet received beyond this jitter threshold.

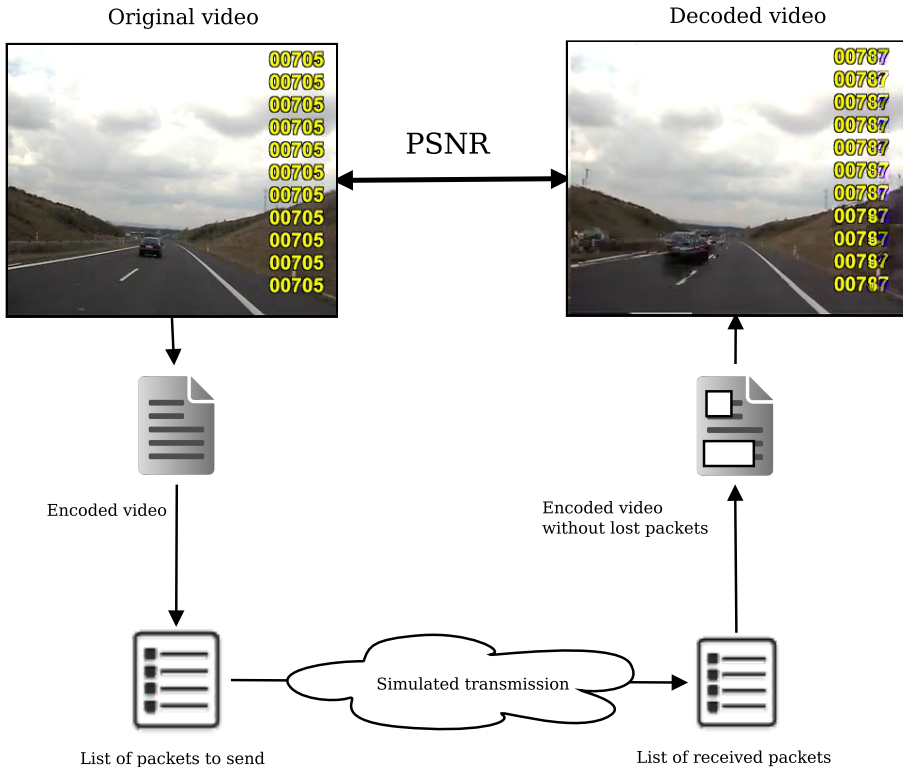


Figure 8.4: Overview of the video quality measurement.

Figure 8.4 shows the process followed to obtain the PSNR.

The first step is to encode the original video using the H.265 video codec and extract the list of packets to be sent. Every packet is a tuple that contains the transmission time, a packet id, and the size of that packet ($\langle time, id, size \rangle$).

After simulating the transmission, we delete from the encoded video sequence those packets that have not been received, thus obtaining a file of the encoded video with some gaps. That file is decoded to obtain a degraded video sequence. For decoding, we employed a modified version of the reference decoder (HM 9.0) presented by Piñol et al. [42], which is able to decode H.265 sequences in the presence of losses.

The degraded video obtained is compared against the original video to obtain the PSNR.

In order to help readers to appreciate the real quality obtained at each PSNR level, some sample videos are made available for visualization and download (<http://www.grc.upv.es/MONET-2014.html>).

8.4.2.2 GPS Emulation

To achieve a realistic scenario, two different types of errors have been introduced.

The first is caused by GPS precision. That is, when the vehicle receives a new position, it has a certain error, and so the emulated position is set within the range of the real position.

The second is caused by the GPS update interval. Real GPS devices update the position periodically (typically 1s). To emulate this behavior, the application level has access only to the estimated position of the last GPS update.

Figure 8.5 shows the difference that exists between the actual vehicle position and the position received at the application level, evidencing the error introduced.

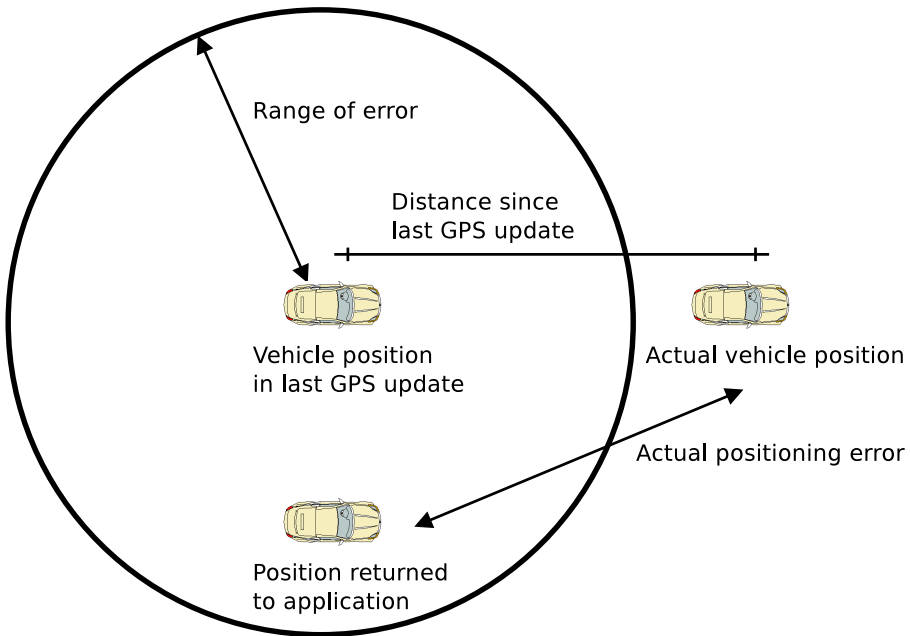


Figure 8.5: GPS error emulation.

8.4.2.3 Simulation Setup

The different flooding schemes have been tested using three vehicle density values. In the highway scenario these densities are achieved by setting an exponential Inter-Arrival Time of vehicles equal to $0.25s$, $1s$, and $3s$. These Inter-Arrival Times correspond, approximately, to a mean density of 100 , 30 , and 10 *vehicles/km*. In the urban scenario we also define three densities: 50 , 100 and 200 *vehicles/km²*. These three densities have been labeled as “High”, “Medium”, and “Low” densities in the figures of this paper.

In addition to every single algorithm and vehicular density setup, we tested all the possible combinations between C (number of extra copies received to stop resend) and *Waiting Time* (maximum time

a node waits to resend a packet). Possible C values are 1, 2, 3, 4, and 5; *Waiting Time* values can be of 100, 200, 300, 400, and 500 *ms*.

The different results presented correspond to the best possible configuration of every algorithm, for each density, in terms of parameters C (number of copies to stop resending) and *Waiting Time* (maximum time that a node waits before resending a message). The optimal parameters for the selected schemes are listed in Tables 8.3 and 8.4. Other schemes have its own optimal parameters but are not listed in order to maintain a small table size. All optimal values for the urban scenarios have a *Waiting Time* of 500*ms*, while the C value is 2, 4, and 5 for “High”, “Medium”, and “Low” vehicular densities respectively.

Non-optimal values can lead to a performance decrease that ranges between 3% and 90% in the worst cases.

In the case of the proposed flooding scheme (ACDB), where no parameters are fixed, the values are chosen in real time by every node based on local density information.

Table 8.3: Parameters for best-case simulations without positioning error in highway scenario.

Density	High		Medium		Low	
	C	Time (ms)	C	Time (ms)	C	Time (ms)
Backfire	2	300	3	200	5	300
Distance Based	1	300	2	500	2	500
Candidate Stability	1	500	1	300	3	300

Table 8.4: Parameters for best-case simulations with positioning error in highway scenario.

Density	High		Medium		Low	
	C	Time (ms)	C	Time (ms)	C	Time (ms)
Backfire	2	300	3	500	3	200
Distance Based	1	300	1	300	3	500
Candidate Stability	1	500	2	300	3	300

8.5 Results

In this section we present experimental results, obtained using the simulation setup and methodology described above.

8.5.1 Comparison of Flooding Schemes

The top graph in Figure 8.6 shows the percentage of received packets for all the different flooding schemes under high vehicle density, while the bottom graph shows the PSNR obtained for the selected flooding schemes. We can see that most schemes are able to provide more than 92% of the packet delivery ratio at a distance of $5km$ from the accident. We can also notice that ACDB is able to improve the packet delivery ratio between kilometers 1 and 4. The Backfire Scheme outperforms the others at the distance of $5km$. This is due to the reduced number of collisions produced when this scheme is used. In contrast, if we focus on the performance from kilometers 5 to 10, we can notice that the Backfire Scheme performs slightly worse compared to kilometers 0 to 5; this is due to a lower vehicle density.

Overall we can see that the DECA-Based Scheme is the worst among the presented algorithms. Due to the high requirements of

video traffic in terms of bandwidth, a high number of collisions occur when vehicles piggyback the information of the received packets. Notice that if vehicles receive information about the packets lost by the neighbors, and they schedule a high number of retransmissions, the wireless medium eventually collapses.

In terms of PSNR, the differences between the selected algorithms are very noticeable. ACDB achieves more than $34dB$ for a distance up to $4km$, while the rest of the algorithms are not able to maintain this level of quality. All the proposed schemes, except DECA, are able to maintain the quality level above $30dB$, a value we consider as the lower bound for a proper video visualization.

Figure 8.7 shows the results for the medium density scenario. In this scenario, the ACDB Flooding Scheme is able to provide the best results up to $4km$, achieving a delivery ratio of more than a 90%. If we focus on the candidate approach, we can see that the Candidate Stability Scheme is able to perform slightly better than the rest, evidencing the importance of a proper selection of the candidate to resend each packet.

With this density, the behavior of the DECA-Based Scheme improves but it still remains highly compromised.

Results for the low density scenario are shown in Figure 8.8. No flooding scheme evaluated in this article is able to provide a good percentage of received packets when the density of vehicles is low. If we focus on the PSNR results, we can see that the received video is unable to achieve $30dB$ at any distance.

8.5.2 Impact of GPS Drift

As described in Section 8.2, some authors state that the errors in the positioning system can affect the reliability of the different GPS-based algorithms.

For this purpose, we tested a feasible case of positioning error. The difference between the real position and the GPS stated position is in a range of $10m$, and the frequency of the position updates is set to $2s$. This configuration can lead, in a worst case scenario to a deviation of about $85m$.

Figure 8.9 shows the results for the high density scenario. Notice that all flooding schemes remain mostly unaffected by the positioning error thanks to the high vehicle density, which is able to compensate for this error.

Figure 8.10 shows the percentage of received packets for the medium density scenario. In this scenario, the different schemes lose a bit of efficiency. However, the ACDB Scheme is still able to maintain a video quality above $30dB$ at a distance of $3km$.

In general terms, the results show that the implemented flooding schemes do not require a precise GPS.

As stated in the previous case, for the low density scenario shown in Figure 8.11, none of the different flooding schemes is able to achieve a percentage of received packets higher than a 90%.

We can notice that the Backfire Scheme suffers from a severe degradation in terms of packet arrival ratio. All other flooding schemes are able to maintain similar packet delivery ratios, with only minor differences among them.

Despite achieving packet delivery ratios higher than 60%, the different flooding schemes are unable to provide a good quality in terms of PSNR; in particular, the results achieved by the different schemes remain below the $30dB$ mark even at a distance of just $1km$.

For a better visual understanding, figure 8.12 shows results for both Backfire and ACDB flooding schemes with and without positioning error. While ACDB results remain mostly the same, the Backfire

scheme suffers from higher losses; nevertheless, it is able to deliver more than 85% of the video packets at a $5km$ distance. The results shown correspond to the medium density scenario.

8.5.3 Urban scenario

The selected scenario, Milan's downtown, is one of the hardest for vehicular transmissions due to the existence of non-straight narrow streets that impede a proper message propagation.

For this set of experiments we selected the most representative flooding schemes according to previous experiments.

We find that, compared to the highway scenario the packet loss distribution is completely different. Figure 8.13 shows the Cumulative Distribution Function (CDF) for both scenarios. While the losses occurring in the Highway scenario were only due to channel contention and collisions, being distributed throughout the simulation time. The urban scenario shows more burstiness in the packet loss due to the existence of buildings.

Figure 8.14 shows the results for the highest vehicular density on the urban scenario. All the tested flooding schemes, except DECA, are able to deliver more than a 90% of the packets at a distance as far as $1.5km$. Focusing on the PSNR chart, we can see that, although ACDB is not able to deliver all the packets to nearby nodes, the behavior in long distances is as good as for the Candidate Stability flooding scheme.

Figure 8.15 shows the behavior of the flooding schemes when the vehicular density is not high. The top chart shows that the delivery ratio of the different flooding schemes is lower than 90% when the distance is greater than $700m$, causing the PSNR to drop below the minimum quality of $30dB$. Regarding the behavior of each flooding scheme, the Distance Based and Candidate Stability schemes,

when properly configured, are able to deliver 5% more packets than ACDB, and 10% more packets than the Backfire scheme. Despite performing considerably better than in a highway environment, the DECA flooding scheme is not able to deliver enough packets for a good video quality at any distance.

Figure 8.16 shows the obtained results when the vehicular density is low. Similarly to the highway environment, all flooding schemes are unable to deliver enough packets to obtain video with a minimum quality level.

Concerning the end-to-end delay experienced in all the different simulations, we can state that every received packet which was not discarded by the nodes, experiences an end-to-end delay that is in all cases lower than 5s.

Overall, we find that vehicle density is a key factor to enable real time video flooding in VANETs. Also, by taking into account the behavior of DECA, we find that flooding schemes that are intended to increase the reliability by using a piggyback system are unsuitable when the traffic load is high. Also, we find that the proposed ACDB Scheme is able to perform adequately without any special requirements in terms of configuration and parameter tuning. This is especially true for highway scenarios, while further improvements should be made to optimize its behavior for urban environments as well.

To sum up, the DECA flooding scheme, due to its piggybacking system, is unable to fulfill all the requirements for a real time video transmission. Candidate-Based flooding schemes have the advantage of being able to select the group of next-hop nodes, thus providing a performance enhancement if adequately selected; however, if they are not properly selected, these schemes introduce an additional delay that reduces the effective throughput. Basic schemes (Counter Based, Distance Based, and Backfire) provide

a good flooding efficiency with an easy control mechanism. The main disadvantage of the Backfire scheme is the excessive packet discarding that requires rising up the number of retransmissions to achieve the same level of performance than other solutions. Finally the ACDB scheme provides a zero-configuration flooding scheme with a good delivery ratio within the established time bounds.

8.6 Conclusions

In this paper, we analyzed ten different flooding schemes for real time video transmission in vehicular environments, providing descriptive metrics such as packet delivery ratio and PSNR for highway and urban scenarios. Additionally, we evaluated the different schemes under a severe positioning error, evidencing that the presented schemes are not significantly influenced by the accuracy of positioning data in highways.

Results show that vehicle density is a key factor when live video streaming is attempted. In fact, under low vehicle densities, video streaming under acceptable conditions is not possible, even at low distances. With a medium density of vehicles, live video streaming in highways has an acceptable video quality up to $4km$, even when imprecise positioning data are used. With a high vehicle density, a good video quality can be achieved for distances up to $5km$. Concerning urban scenarios, a good real-time video flooding can only be achieved with a high density of vehicles.

Based on a wide set of experiments under different conditions, we find that the proposed Automatic Counter Distance Based (ACDB) Flooding Scheme is able to achieve a high percentage of delivered packets within low delay bounds without any special configuration requirements, and that it is able (in most of the cases) to outperform the majority of the flooding schemes tested even when they

are optimally configured.

8.6 Conclusions

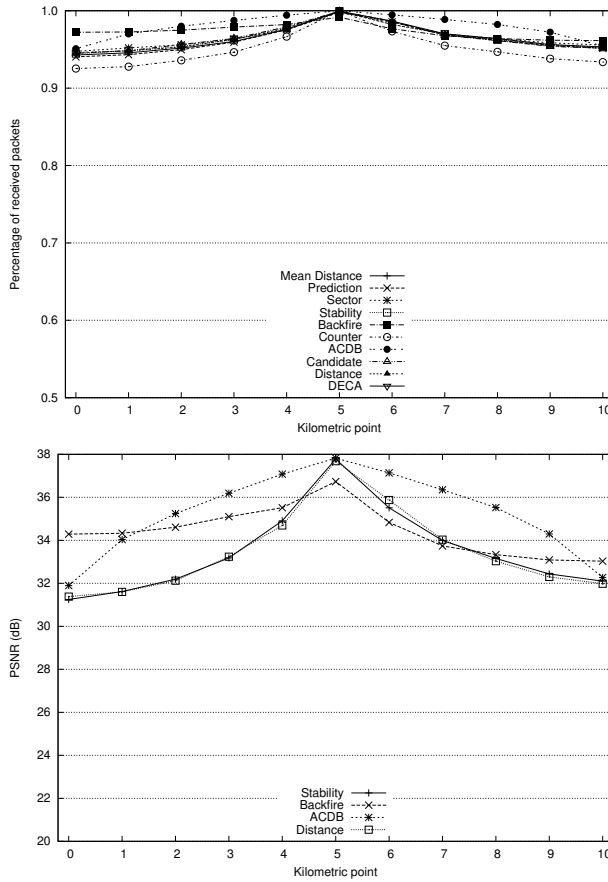


Figure 8.6: High density scenario. Percentage of packets received (up) and PSNR (down).

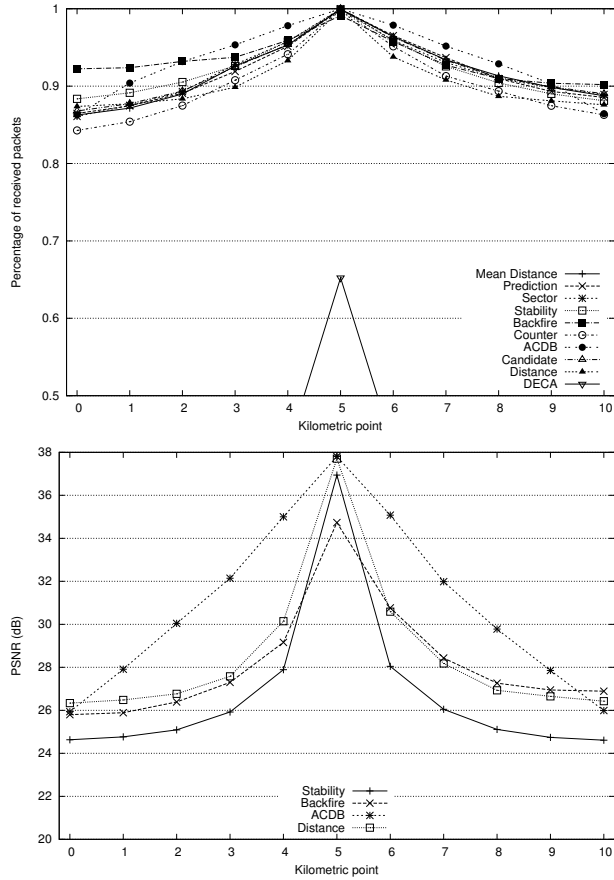


Figure 8.7: Medium density scenario. Percentage of packets received (up) and PSNR (down).

8.6 Conclusions

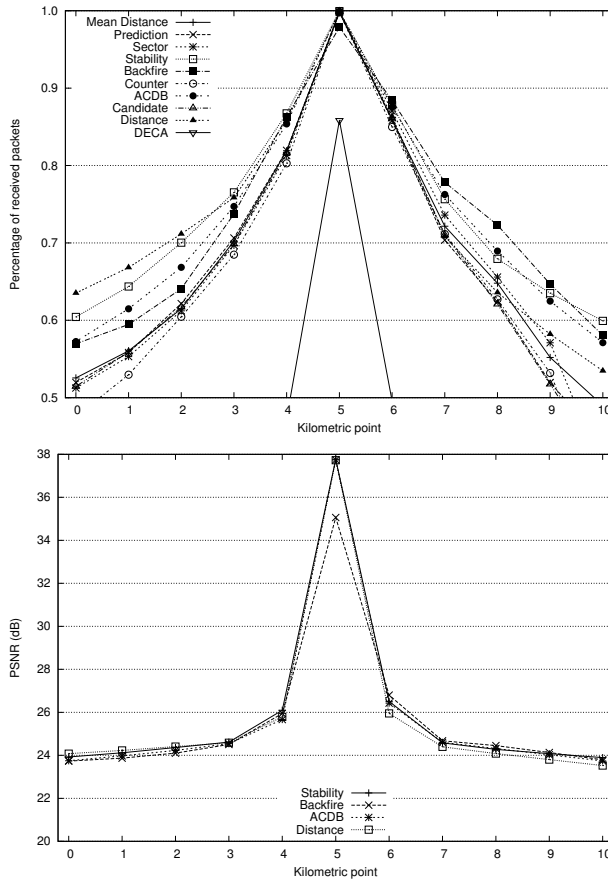


Figure 8.8: Low density scenario. Percentage of packets received (up) and PSNR (down).

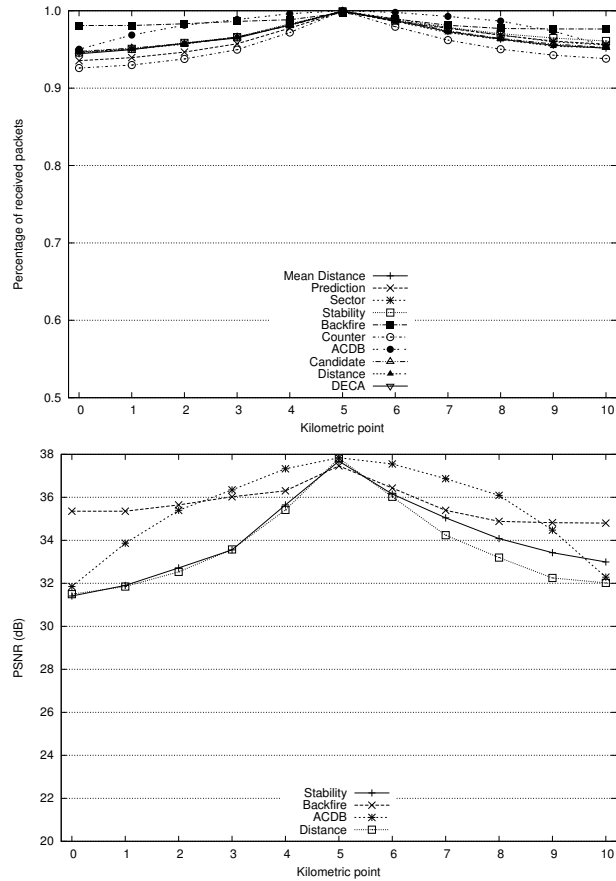


Figure 8.9: High density scenario. Percentage of packets received (up) and PSNR (down).

8.6 Conclusions

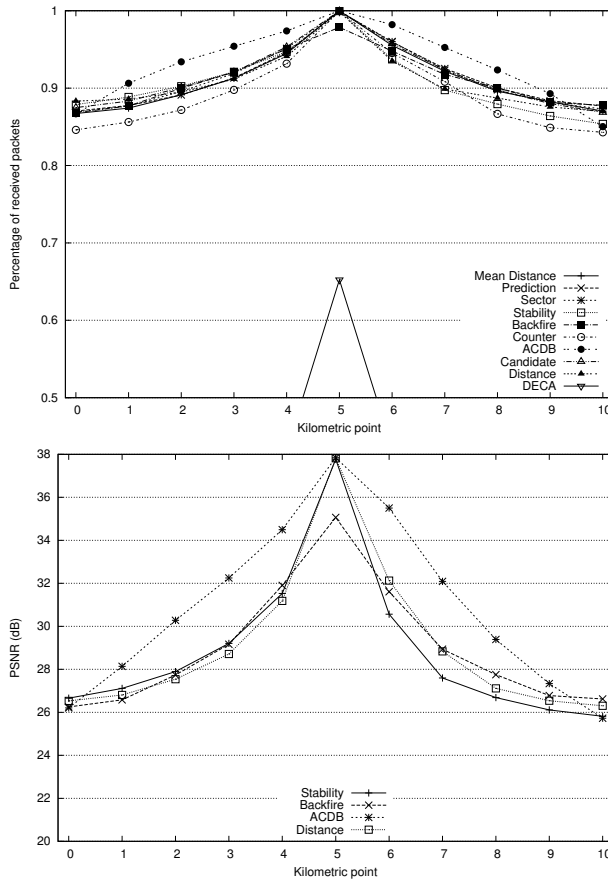


Figure 8.10: Medium density scenario. Percentage of packets received (up) and PSNR (down).

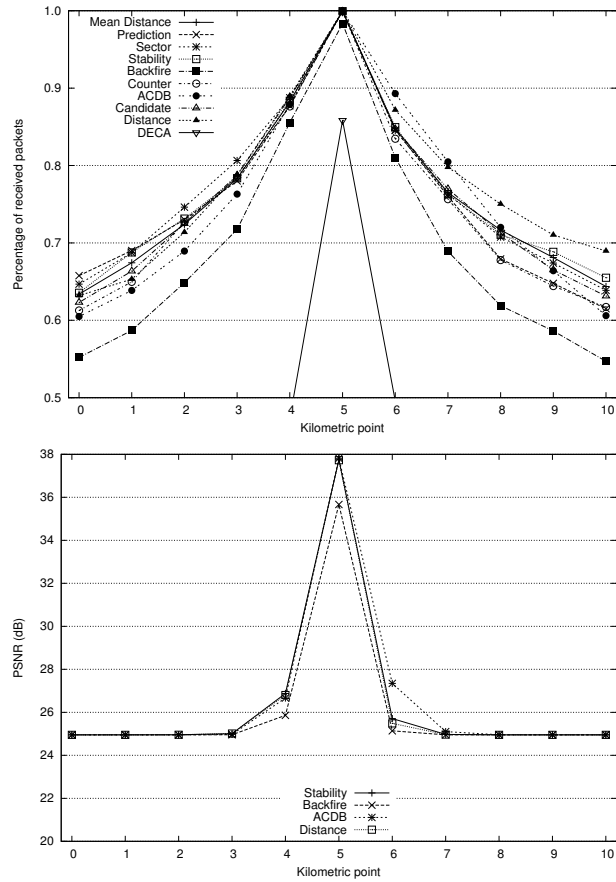


Figure 8.11: Low density scenario. Percentage of packets received (up) and PSNR (down).

8.6 Conclusions

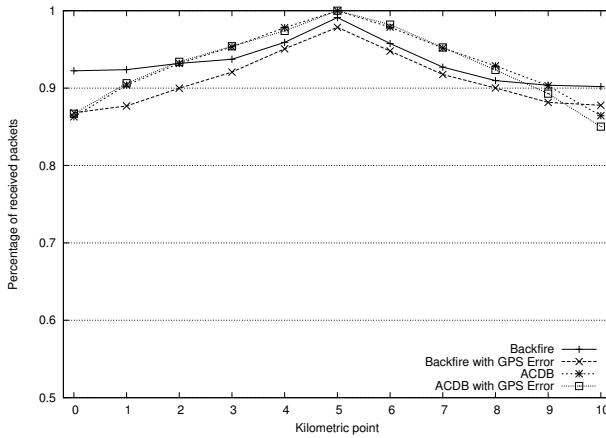


Figure 8.12: Medium density scenario. Percentage of packets received. Positioning error comparison.

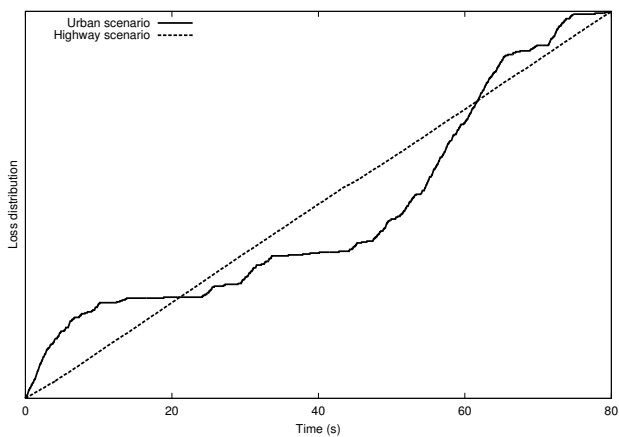


Figure 8.13: Cumulative Distribution Function for packet loss in both scenarios.

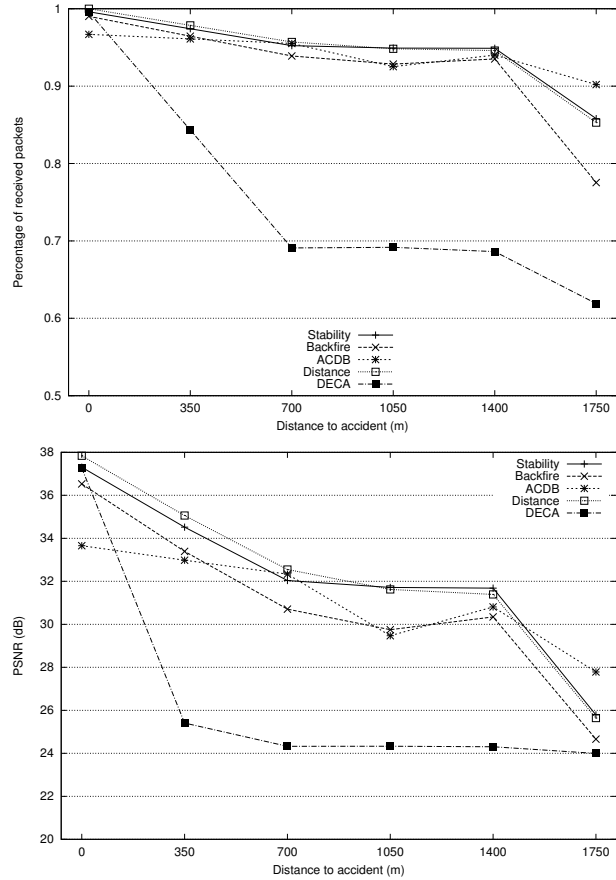


Figure 8.14: Urban High density scenario ($200\text{vehicles}/\text{km}\hat{A}^2$). Percentage of packets received (up) and PSNR (down).

8.6 Conclusions

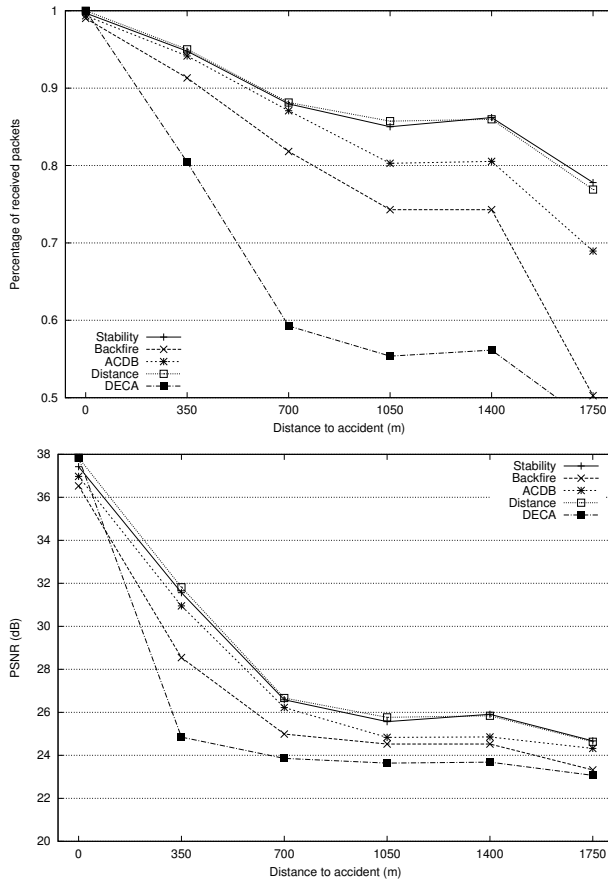


Figure 8.15: Urban Medium density scenario ($100\text{vehicles}/\text{km}^2$). Percentage of packets received (up) and PSNR (down).

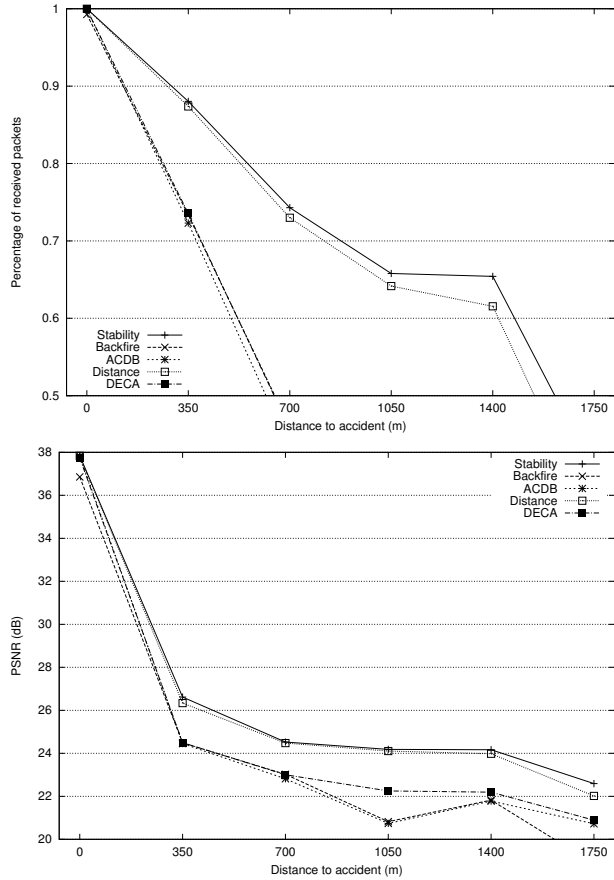


Figure 8.16: Urban Low density scenario ($50\text{vehicles}/\text{km}\hat{A}^2$). Percentage of packets received (up) and PSNR (down).

Results' discussion

In this thesis our focus is on reliable video delivery in wireless environments.

First we focused on peer-to-peer video, typical of MANET environments where two endpoints want to communicate without requiring any infrastructure. Such video delivery mode, applicable for example to communication between members of a rescue or firemen team, imposes strong QoS requirements on the MAC layer, also requiring a distributed admission control system to determine end-to-end conditions, in addition to other requirements (efficient routing mechanisms, efficient video coding, etc.). So, the next two blocks focus on these issues. In particular, the first block discusses MAC layer QoS, determining its effectiveness in real world environments, and emphasizing on performance differences between simulations and real conditions, which should be taken into consideration by researchers working in the field. The second block addresses the challenges of developing, deploying and testing a real implementation of a distributed admission control system, again highlighting the differences between simulated and real conditions,

and proposing alternatives to improve performance when simulation expectations are not met.

Once the study of unicast video delivery was completed, we instead focused on multicast video distribution in wireless multi-hop environments. Such video data can be quite useful in vehicular environments, where video about an accident can be delivered to multiple vehicles and relayed to the authorities in a multi-hop fashion. This case differs significantly from the previous unicast case since now data frames are transmitted via broadcast and not unicast. This means that MAC-based QoS solutions as provided by the IEEE 802.11e annex are no longer applicable, neither are the admission control techniques used for unicast video delivery. So, alternative dissemination strategies had to be sought in the scope of this thesis, aiming at mitigating the broadcast storm problem that usually arises in wireless environments when the channel experiences intensive broadcasting by several competing sources.

Below we proceed to describe in more detail the three blocks defined above, referring the findings and contributions of the different related publications.

Assessing the effectiveness of MAC-level QoS mechanisms in real-world environments.

The proliferation of devices with multimedia and wireless networking capabilities has created an interest in supporting Quality of Service (QoS) in ad-hoc networks.

To meet these needs, the IEEE 802.11e working group enhanced the IEEE 802.11 standard to provide QoS at the MAC level. Having a good MAC level QoS is fundamental in contention-based

wireless networks to achieve traffic differentiation in terms of both throughput and delay.

The IEEE 802.11e protocol has been previously studied in simple scenarios where an access point and several stations communicate, but more complex scenarios where several infrastructureless ad-hoc networks are involved have not been studied in real-world situations.

The first block includes papers A and B, and provides valuable data through a real world experiment. This experiment is quite useful for the research community as there are not many studies with real devices, being almost all the research in the field being done in simulated scenarios.

Publication A presents a testbed to compare IEEE 802.11e simulation results with a real-world scenario. The real-world scenario consists of four source-destination pairs, as well as three intermediate stations, thus conforming a total of eleven computers.

The simulation results used for comparison were obtained using the ns-2 simulator, configured with the same 4-hop scenario.

First we realize a traffic differentiation analysis, that is, for each source-destination pair we gradually increase the traffic load to evaluate the effectiveness of the QoS mechanism. Due to the lower channel capacity of the testbed environment, the throughput for all the traffic types decays more abruptly when the saturation point is reached. Additionally, the traffic differentiation effectiveness is partially lost. Concerning the delay results, the obtained values experience a much more steep growth for low load values, and when moderate/high saturation levels are reached the differentiation capabilities of IEEE 802.11e experience a significant loss in effectiveness compared to simulation results. For the sake of clearness we have replicated these charts in figures 9.1 and 9.2.

The second experiment was the QoS robustness analysis, where the

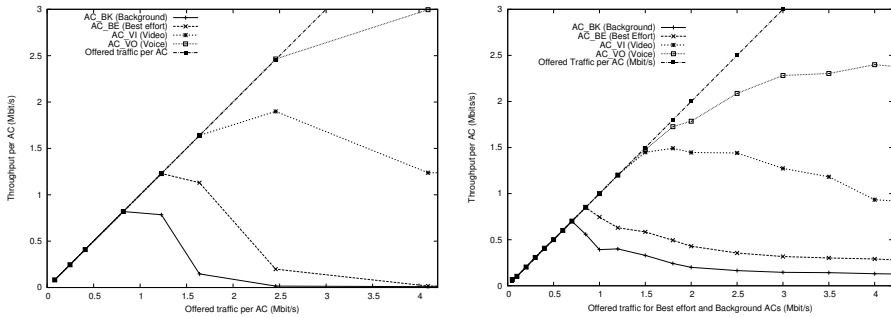


Figure 9.1: Throughput values achieved via simulation (left) and in our testbed (right).

Video and Voice traffic rate was fixed while the Background and Best Effort traffic rates were gradually increased. Results show that in the testbed, and contrarily to simulation results, video and voice traffic experience some degradation. In terms of end-to-end delay, when the saturation point is reached, video and voice traffic experience an increase of about two orders of magnitude compared to simulation results (Figures 9.3 and 9.4).

To conclude, obtained results show that, despite the differences between the simulated environment and the testbed, we can find similar overall trends in both scenarios. In general, we find that simulation-based results tend to be optimistic compared to real-world performance.

Publication B extends the work done in Publication A. In the previous work all the nodes were placed within one meter and without physical barriers that could fade the wireless signal, thus enforcing a multihop topology using Linux iptables application.

In order to create a more realistic scenario, eight computers were deployed throughout the floor of an university building, enforcing multi-hop connectivity in a natural manner due to distance and

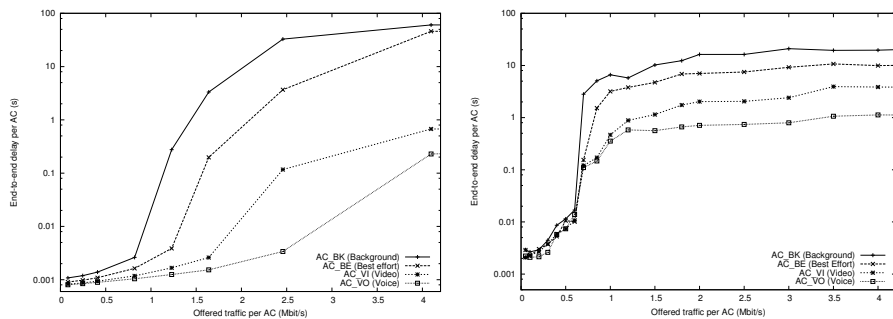


Figure 9.2: End-to-end delay values achieved via simulation (left) and in our testbed (right).

the presence of brick walls, wooden doors, double glass windows and metal filling cabinets in each room.

As shown in Publication A, a Fast-Ethernet wired network is used for station management and control purposes.

In these experiments we evaluate the QoS conditions experienced by a bidirectional videoconferencing session that takes advantage of the video and voice MAC access categories of IEEE 802.11e. Additionally, to evaluate the influence of the Best Effort traffic, we start an FTP/TCP flow between both endpoints involved in videoconferencing.

Due to the presence of intermittent and low quality links between pairs of nodes we compared static routing versus dynamic routing. Static routing was configured to make use of the short and stable links even though it implies having a large number of hops when two nodes communicate, while in the dynamic routing scenario the well-known OLSR protocol was used.

In both scenarios both throughput and delay were measured for all possible origin/destination pairs from one hop up to four hops in the static scenario (since OLSR is a dynamic protocol we cannot

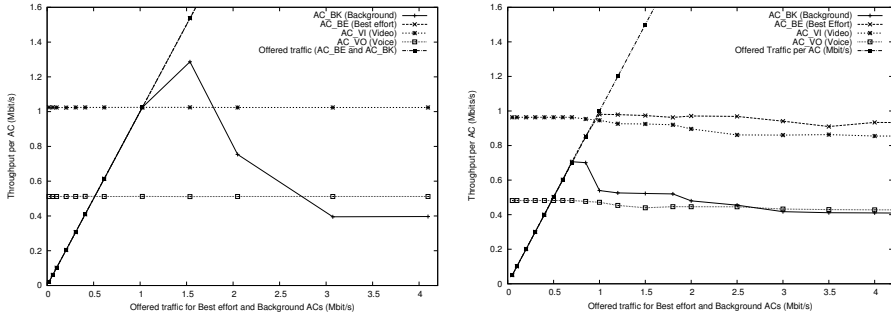


Figure 9.3: Throughput stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).

know the exact number of hops).

Results show that the IEEE 802.11e standard is able to maintain its effectiveness even in a real multihop scenario, as long as the quality of the links is good. On the contrary, dynamic protocols tend to employ the shortest route and, therefore, favour low quality links.

Figure 9.5 shows results for the three-hop case.

In general, we find that dynamic routing is prone to cause intermittent connectivity problems that cause problems to both QoS traffic (that experiences data loss and increased delay) and TCP traffic (that suffers from starvation).

Mechanisms for efficient unicast video transmission.

The second block is the work done in the unicast scenario, that is, when there is a fixed destination for the video sequence. Although

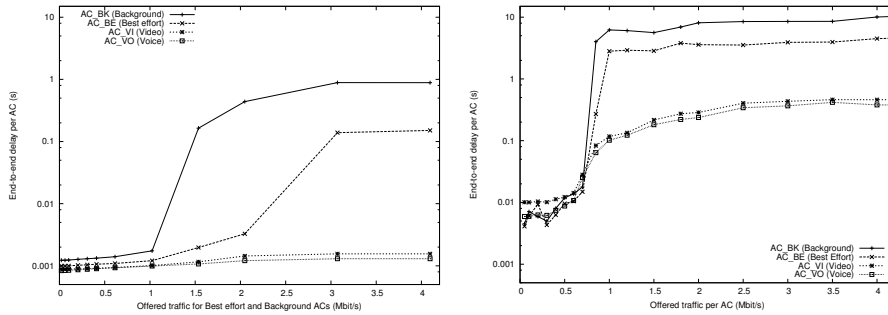


Figure 9.4: End-to-end delay stability for voice and video traffic when increasing best effort and background traffic. Data from simulation (left) and from the testbed (right).

the developed solution targeted a MANET environment, all the obtained data is applicable to vehicular scenarios too.

MANETs are networks where real-time services such as videoconferencing or VoIP are not easy to provide due to the topology of the network. This type of networks lacks a good Quality of Service (QoS) support. In this scope, the IEEE802.11e working group provided QoS support at the MAC level, although is still not enough when facing QoS flow concurrency.

One of the most crucial components of a system attempting to provide QoS guarantees is the Admission Control Module.

In Publication C, after studying the state of the art in Admission Control Modules, we decided to implement and test DACME (Distributed Admission Control for Manet Environments) in a real-world scenario. DACME, contrarily to most existing solutions, does not require a modification of the MAC layer, and the required resources for running it are low. Additionally, due to its end-to-end design, it does not impose any restriction on intermediate nodes.

For testing the effectiveness of DACME we created a seven-hop

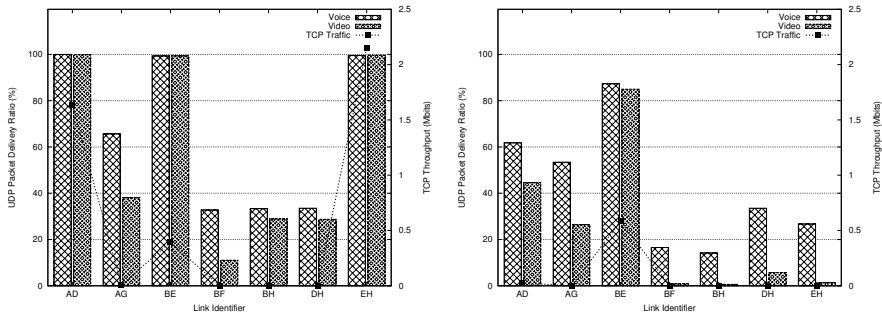


Figure 9.5: UDP packet delivery ratio (Video and Voice) and TCP throughput (Best effort) for all source-destination pairs at three hops using static (left) and dynamic (right) routing.

ad-hoc network, which allows us testing with different hop number combinations per flow.

To manage the repeatably of the experiments we modified Castadiva, a java-based MANET emulator, to be able to manage tests with DACME support.

In the experiments we have three concurrent video flows and, for each test, we increase the number of best-effort flows (from zero to four) in order to assess the impact of congestion on QoS performance.

Results (figure 9.6) showed a great improvement both in throughput stability and reduced delay when comparing the usage of DACME against not using DACME.

So, with this paper we conclude that IEEE 802.11e alone is not enough to provide QoS management in a MANET environment, being an Admission Control mechanism essential for achieving proper QoS levels. Additionally, results also show that the DACME architecture was fully effective in a real testbed.

Following the experiments that highlighted an adequate behavior

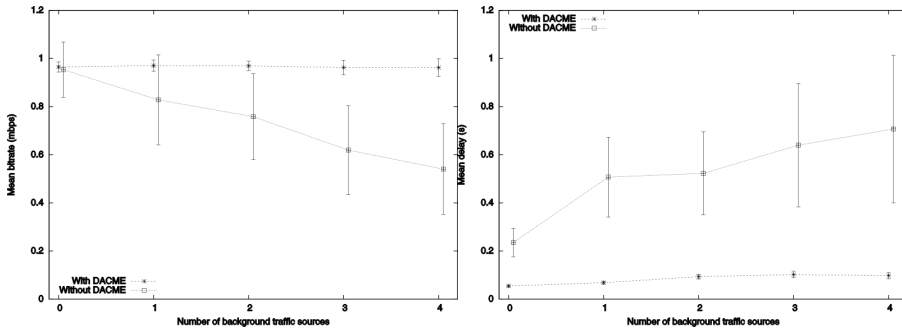


Figure 9.6: Mean bit rate when flows are active (left) and mean end-to-end delay (right).

of our DACME implementation, as shown in Publication C, we detected some differences between simulation and real-world scenarios.

The Admission Control Module (ACM) should be able to estimate the resources of the network and decide when application flows should be admitted or rejected, avoiding to interfere with previously active flows. Unfortunately, this is not an easy task since MANETs are highly dynamic environments, and thus flow admission does not guarantee good QoS conditions throughout time.

With the data obtained from several tests, and based on the acquired experience, in Publication D we propose an enhancement to the DACME decision module which offers better adaptability to the network bandwidth fluctuations under wireless interference conditions.

DACME learns about the network status by using a probe/response strategy. In particular, to achieve an accurate bandwidth estimation, DACME uses a burst of probe packets periodically generated by the source in a back-to-back fashion. These packets arrive to the destination node with an average inter-packet time gap which

allows the destination node to make an estimation of the available end-to-end bandwidth. When the destination gathers all the data required, it sends a response packet with the current bandwidth estimation (BM) back to the source. DACME relies heavily on bandwidth estimations to decide whether to admit or deny a flow.

The original DACME bandwidth estimation algorithm, called Confidence Interval Based Decision Algorithm (CIB-DA), worked extremely well in simulated scenarios, but in real-world scenarios, due to the inaccuracy of the timers, and also to some hardware and software limitations, we obtained some unexpected bandwidth measurements. These measurements led to frequent non-desired flow interruptions.

To overcome the problems experienced in real-world scenarios, we modified CIB-DA to create Hybrid Range / Confidence Interval Decision Algorithm (HRCI-DA).

HRCI-DA was tested in three different scenarios. In the first set of experiments we study how best effort traffic affects the stability of video flows. As shown in Publication C, we have three concurrent video flows, and for each test we increase the number of best-effort flows from zero to four.

The second set of experiments allowed assessing the performance of our implementation of DACME when handling a variable number of QoS flows, as well as analyzing the interactions between these flows. With this purpose our experimental settings are similar to those of the previous experiment, but we now fix the number of best effort flows to three, while increasing the number of video flows from one to four (see figure 9.7).

In both cases results show that IEEE 802.11e alone is not powerful enough to provide QoS guarantees in MANET scenarios, while, when using DACME, video flows experience a better QoS. If we compare both DACME decision algorithms both have similar val-

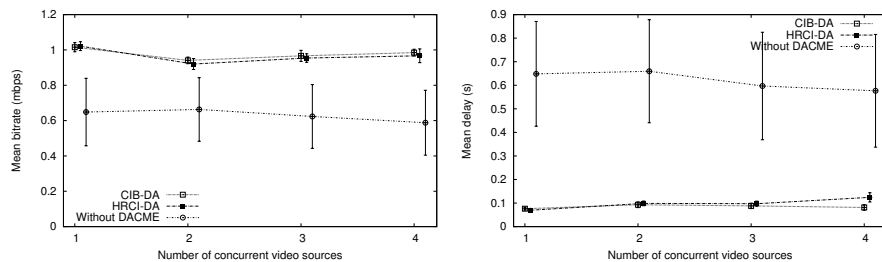


Figure 9.7: Mean values for throughput (left) and end-to-end delay (right) when varying the number of concurrent video sources.

ues in terms of throughput and delay, but the new HRCI-DA is able to maintain flows active for longer periods with less user interruptions (see figure 9.8).

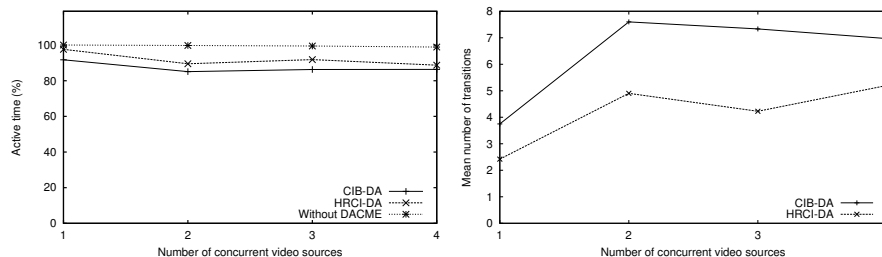


Figure 9.8: Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the number of concurrent video sources.

Finally, the goal of the third set of experiments is to test the behavior of DACME when varying the maximum delay allowed for the video flows to determine the degree of compliance achieved. Both decision algorithms achieve the same delay compliance, which is very good when the required delays are above $150ms$, but both algorithms experience the same problems when the maximum delay is set to $100ms$ (see figure 9.9).

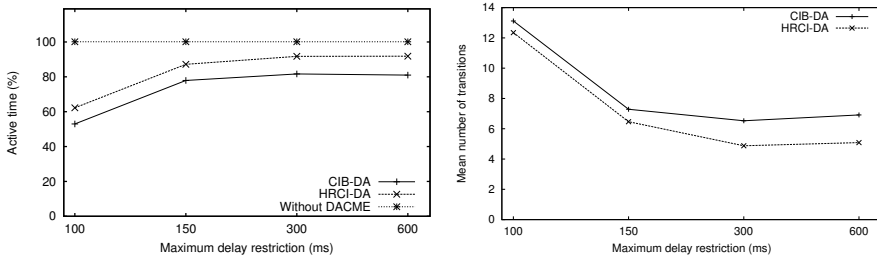


Figure 9.9: Mean active time per flow (left) and mean number of state transitions for DACME flows (right) when varying the maximum delay restriction.

To sum-up, in Publication D we proposed an improvement to DACME that boosts its performance in real-world scenarios, and allows users to have improved unicast video transmissions.

By adopting our solution, good end-to-end video transmissions in MANETs can be achieved despite of relying on wireless multi-hop transmissions, and despite of other sources of competing traffic (both QoS and best effort traffic).

Video broadcasting in VANETs

In a near future it is expected that vehicular networks (VANETs) will be as extended as mobile phones are nowadays. Vehicles will be equipped with different sensors which can provide useful information to other drivers or traffic authorities. One of the most useful sources of information a vehicle can provide is the live video of an accident, not only allowing the emergency services to know in advance the exact status of the accident, but also to help other vehicles to decide whether to change their current route.

Unfortunately, VANETs provide one of the most difficult environ-

ments to achieve a good transmission quality since this type of networks involves high relative speeds, which cause short connection times and transmission problems such as the Doppler effect. In addition, bandwidth is typically very limited, thus becoming one of the worst-case scenarios for real-time video transmission.

This third and last block represents a complete change in terms of both methodology and scenario compared to the former blocks since all of the results were obtained using simulation tools and not real devices like in the previous ones. This change was due to the high cost and complexity associated with testing using real vehicles.

Publication E studies the feasibility of achieving real-time video transmission with current broadcasting techniques, also evaluating the final quality of the obtained video. Additionally, with the recent approval of the new H.265 video compression standard, we evaluate the differences towards the H.264 standard.

The simulated scenario consists of a two-lane, single-way, 10 *km* long highway, and the intended information flow is generated in one vehicle stopped on the middle of the highway that broadcasts video to all other vehicles. Quality measurements are made at every kilometer.

Two flooding algorithms were implemented: Counter-Based and Distance-Based. The Counter-Based flooding scheme relies on the concept of every vehicle having an internal count of the number of times that a certain packet is received. If the packet has been received less than a certain number of times, let's say C , the vehicle resends it after a random time wait. On the other hand, if the packet has been received more than C times, then the vehicle just discards it.

The Distance-Based flooding scheme is similar to the Counter-Based one, as it relies on the counter mechanism, but the random

time wait is not random anymore; instead it now depends on the distance to the sender vehicle. The further it is the lowest the waiting time.

The different flooding algorithms implemented require a fine tuning, having each of them two parameters for this purpose. In Publication E the first set of results were employed to select the best parameter configuration for different vehicular densities.

The second set of results shows the differences between video coding algorithms when using the selected configurations of the flooding algorithms in order to select the best codec/flooding algorithm combination. Focusing on the differences between codecs, when employing the same flooding configuration, the H.264 compressed video has a lower packet arrival rate than the H.265 one due to the higher bitrate that H.264 injects into the network. (Figure 9.10)

Finally, the third set of experiments shows differences in terms of quality and error resilience for both video codecs. Graphs in Publication E highlight that H.265 outperforms H.264 in terms of both frame loss and PSNR quality. From the user perspective, the video flooded using the H.265 codec is able to provide a much smoother experience than H.264 encoded videos.

As Publication E presented very limited and preliminary results, in Publication F we improved the scenario. Now the highway was more realistic as we used a two-way rather than a one-way highway..

The flooding algorithms used in Publication E (Counter-Based and Distance-Based) were similar and simple, so we adapted and implemented two new algorithms. Backfire is based on the Distance-Based approach but it includes aggressive packet discarding techniques to reduce network traffic. The second algorithm, DECA, is based on a piggybacking principle to ensure data packet delivery.

All the different throughput and delay comparisons were made using the best parameters for each flooding algorithm in each sce-

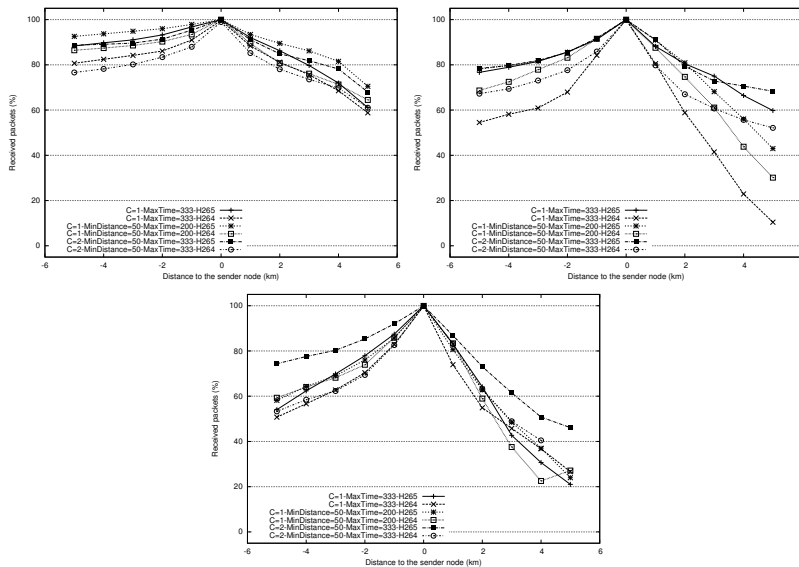


Figure 9.10: Packet delivery ratio. Vehicle arrival rate of 1s (top-left), 2s (top-right), 3s (bottom).

nario.

Results show that the concept of piggybacking employed by DECA is not able to cope with the restraints imposed by real-time video flooding due to the high amount of data packets required and the reduced time to deliver the packets (Figure 9.11).

On the other hand, Backfire showed a very good behavior when vehicular density was high, but not as good when vehicular density was low (Figure 9.12).

Publication G presents a whole set of new flooding algorithms that make use of the beaconing system that every vehicle taking part in a vehicular network must have. With these new schemes we try to achieve a better packet delivery ratio. The main drawback of the flooding schemes presented in publications E and F, as well as

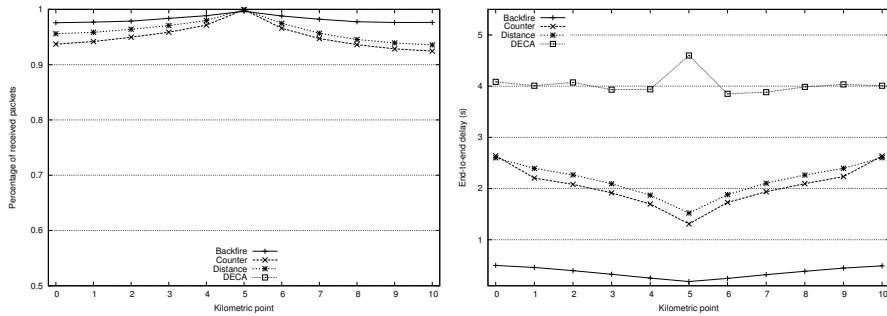


Figure 9.11: High density scenario. Percentage of packets received.

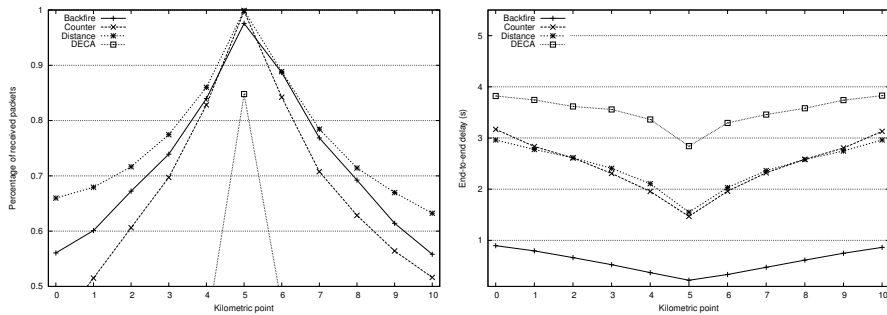


Figure 9.12: Low density scenario. Percentage of packets received (left) and delay (right).

the ones developed for Publication G, is that for achieving a good behavior each scenario requires a different set of parameters. That is why, in Publication G, we also present the Automatic Copies Distance Based (ACDB) flooding scheme. ACDB does not require any configuration, and it is capable of coping with different density scenarios without the need of changing its working parameters.

Results show that ACDB is able to adapt to any vehicular density situation, and at the same time it is able to achieve high packet delivery ratio and PSNR results, showing that real-time video flood-

ing is feasible, even in long distances, if the vehicular density is medium or high (Figure 9.13).

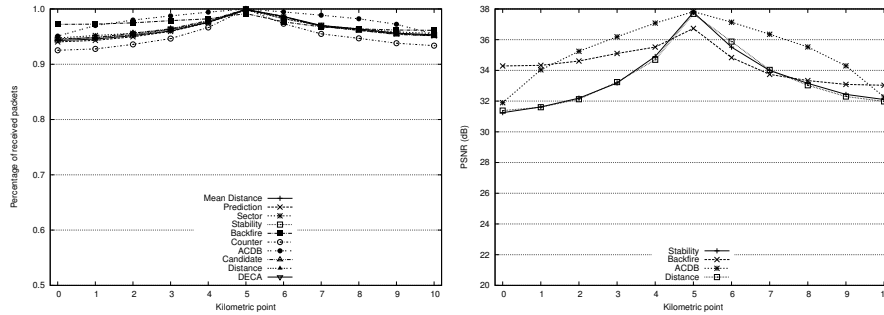


Figure 9.13: High density scenario. Percentage of packets received (left) and PSNR (right).

One of the main drawbacks of Distance-Based flooding schemes is the dependence from a reliable positioning system. In order to measure the loss of effectiveness that the different schemes experience when the positioning system does not give accurate information, we implemented a GPS module with two different sources of error. The first one is positioning error, which in this case was set to a maximum of 10m. The second one is the obsolescence error, where the positioning update only takes place after several seconds, leading to a maximum deviation of approximately 80m when combining both sources of error.

Results show that this error does not affect in a significant way the different flooding schemes implemented, and so, the different flooding schemes are reliable for real-world usage.

Since all the results obtained above targeted highway scenarios, Publication G also includes simulations in an urban scenario. The layout selected was a 5km² map of the city of Milan. The selected area is a very difficult scenario for flooding algorithms as it mainly

contains short and narrow streets with a high number of buildings that make it very difficult to communicate with other vehicles.

As in the highway scenario, the video transmission starts from a vehicle stopped just in the middle of the scenario. Several measurement points were deployed every $350m$, thus achieving 5 different distance ranges.

Results show that, although ACDB is not the best flooding scheme, it is able to provide a good video quality in high vehicular density scenarios, having a behavior comparable to the best alternative solutions for the rest of densities (Figure 9.14).

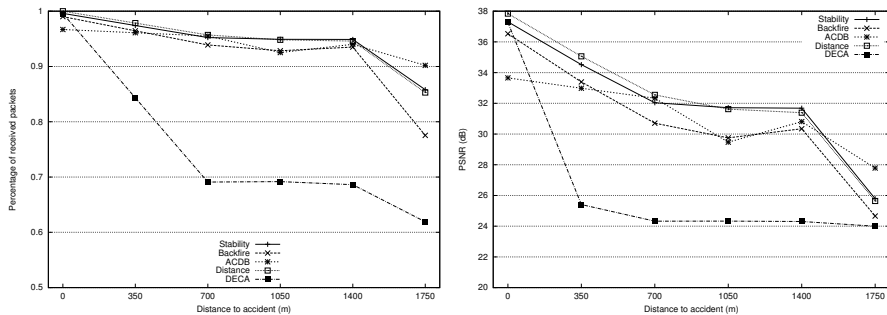


Figure 9.14: Urban High density scenario ($200\text{vehicles}/\text{km}^2$). Percentage of packets received (left) and PSNR (right).

In general, with the proposed flooding schemes, real-time video flooding in VANET environments is feasible if the density of vehicles on the road is medium or high. On the other hand, when the vehicular density is low, real-time video transmission is unfeasible in both, highway and urban scenarios.

10

Conclusions

In the development of this thesis our main focus was on providing a useful framework to achieve real-time video transmission in vehicular environments, both in the theoretical and applied domains.

Currently most advances in terms of MANET / VANET protocols are tested in simulated environments. On the contrary, due to the high costs involved, there is basically no testing in real environments.

In this thesis we have performed detailed comparisons between simulated and real environments, obtaining a valuable experience concerning the limitations of the real-world implementations.

Based on that knowledge, we have been able to propose and improve different algorithms that provide a good basis for real-time video transmission in a broad set of scenarios.

Having accomplished all of the objectives stated in section 1.4, we consider that the purpose of this thesis has been successfully achieved.

In coming years significant improvements are expected in the ve-

hicular ad-hoc networks field as manufacturers start to introduce them in new models.

This thesis will provide a basis for services that require video streaming. This includes services as overtaking assistance, that could use the unicast solutions provided in this thesis.

The multicast solutions presented might be used by traffic authorities for monitoring certain points of the road, as they can make use of the vehicles passing-by without having the need of installing security cameras. This idea opens a new set of possibilities, as several algorithms should be developed in order to obtain the better camera selection to visualize a certain incident.

Concerning future work associated with this thesis, an improvement of ACDB is required to cope better with urban scenarios, as well as the creation of a layer to couple the flooding scheme with the video encoder in order to achieve a better video quality.

Summary of publications

Publication A:

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Deploying a real IEEE 802.11e testbed to validate simulation results*. IEEE 34th Conference on Local Computer Networks. (LCN 2009). Pages 109 to 115, October 2009.

Publication B:

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Assessing the IEEE 802.11e QoS effectiveness in multi-hop indoor scenarios*. Ad Hoc Networks, Volume 10, Issue 2. Pages 186 to 198. March 2012.

Publication C:

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Distributed Admission Control in 802.11e-Based MANETs: from Theory to Practice*. IFIP Wireless Days (WD), 2011. Pages 10 to 12. October 2011

Publication D:

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Evaluating the Effectiveness of a QoS Framework for MANETs in a Real Testbed*. Ad-hoc, Mobile, and Wireless Networks Volume 7363. Pages 221 to 234. 2012

Publication E:

Alvaro Torres, Pablo Piñol, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *Evaluating H.265 real-time video flooding quality in highway V2V environments*. Wireless Communications and Networking Conference (WCNC), 2014. Pages 2716 to 2721. April 2014

Publication F:

Alvaro Torres, Yusheng Ji, Carlos Calafate, Juan-Carlos Cano and Pietro Manzoni. *V2X Solutions for Real-time Video Collection*. 11th Annual Conference on Wireless On-demand Network Systems and Services (WONS). Pages 116 to 121. April 2014

Publication G:

Alvaro Torres, Carlos Calafate, Juan-Carlos Cano, Pietro Manzoni and Yusheng Ji. *Evaluation of Flooding Schemes for Real-time Video Transmission in VANETs*. Ad Hoc Networks, Volume 24. Pages 3-20, January 2015

Funding Acknowledgments

This work was partially supported by:

- the *Ministerio de Educación y Ciencia*, Spain, under Grant TIN2008-06441-C02-01.
- the *Ministerio de Ciencia e Innovación*, Spain, under Grant TIN2011-27543-C03-01.
- the *Ministerio de Educación*, Spain, under the FPU program, AP2009-2415.
- the *Universidad Politécnica de Valencia*, Spain, under project SECURA-3252.
- the *Generalitat Valenciana* under Grant GV/2009/010.
- the *National Institute of Informatics International Internship Program*.

- [1] H.264 standard. <http://www.itu.int/rec/T-REC-H.264>.
- [2] H.265 standard. <http://www.itu.int/rec/T-REC-H.265>.
- [3] INET framework. <http://inet.omnetpp.org/>. Accessed: April 4, 2013.
- [4] MadWifi.net. Available at: <http://madwifi.org/>. Accessed: January 30, 2009.
- [5] MetaGeek Wi-Spy. Available at: <http://www.wi-spy.com/>. Accessed: January 29, 2009.
- [6] OMNeT++ simulator. <http://www.omnetpp.org/>. Accessed: April 4, 2013.
- [7] Ralink Technology Corporation. Available at: <http://www.ralinktech.com/>. Accessed: January 30, 2009.
- [8] RTP Payload Format for High Efficiency Video Coding. <https://tools.ietf.org/html/draft-schierl-payload-rtp-h265-03>. Accessed: May 30, 2015.

- [9] The netfilter.org iptables project. Available at: <http://www.netfilter.org/>. Accessed: July 28, 2011.
- [10] Video Traces. <http://trace.eas.asu.edu>. Accessed: January 15, 2013.
- [11] M. Baguena, C.T. Calafate, J. Cano, and P. Manzoni. Towards realistic vehicular network simulation models. In *Wireless Days (WD), 2012 IFIP*, pages 1–3, 2012.
- [12] M. Baguena, S.M. Tornell, A. Torres, C.T. Calafate, J.-C. Cano, and P. Manzoni. Vacamobil: Vanet car mobility manager for omnet++. In *Communications Workshops (ICC), 2013 IEEE International Conference on*, pages 1057–1061, 2013.
- [13] A. Banchs, A. Azcorra, C. Garcia, and R. Cuevas. Applications and challenges of the 802.11e EDCA mechanism: an experimental study. *IEEE Network*, 19(4):52–58, 2005.
- [14] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz. Sumo - simulation of urban mobility: An overview. In *SIMUL 2011, The Third International Conference on Advances in System Simulation*, pages 63–68, Barcelona, Spain, October 2011.
- [15] C.T. Calafate, M.P. Malumbres, J. Oliver, J.C. Cano, and P. Manzoni. QoS Support in MANETs: a Modular Architecture Based on the IEEE 802.11e Technology. *Circuits and Systems for Video Technology, IEEE Transactions on*, 19(5):678–692, May 2009.
- [16] C.T. Calafate, P. Manzoni, and M.R. Malumbres. Assessing the effectiveness of iee 802.11e in multi-hop mobile network environments. In *Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, 2004. (MASCOTS*

- 2004). *Proceedings. The IEEE Computer Society's 12th Annual International Symposium on*, pages 205–212, Oct 2004.
- [17] J. Cano, J.-C. Cano, Chai-Keong Toh, C.T. Calafate, and P. Manzoni. EasyMANET: an extensible and configurable platform for service provisioning in MANET environments. *IEEE Communications Magazine*, 48(12):159–167, 2010.
- [18] J.-C. Cano, J.-M. Cano, C. Calafate, E. Gonzalez, and P. Manzoni. Evaluation of the trade-off between power consumption and performance in bluetooth based systems. In *Sensor Technologies and Applications, 2007. SensorComm 2007. International Conference on*, pages 313–318, Oct 2007.
- [19] X. Chen, H. Zhai, X. Tian, and Y. Fang. Supporting QoS in IEEE 802.11e wireless LANs. *IEEE Transactions on Wireless Communications*, 5(8):2217–2227, 2006.
- [20] Francesca Cuomo, Izhak Rubin, Andrea Baiocchi, and Pierpaolo Salvo. Enhanced VANET broadcast throughput capacity via a dynamic backbone architecture. *Ad Hoc Networks, Vol. 21, October 2014, pages 42-59*.
- [21] Ian Dangereld, David Malone, and Douglas J. Leith. Experimental Evaluation of 802.11e EDCA for Enhanced Voice over WLAN Performance. In *4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 1–7. IEEE, April 2006.
- [22] A. Derhab and A. Bouabdallah. Admission control scheme and bandwidth management protocol for 802.11 ad hoc networks. In *Innovations in Information Technology, 2007. IIT '07. 4th International Conference on*, pages 362 –366, nov. 2007.

- [23] K. Duffy, D. J. Leith, T. Li, and D. Malone. Improving Fairness in Multi-Hop Mesh Networks Using 802.11e. In *4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 1–8, 2006.
- [24] M. Fogue, P. Garrido, F. Martinez, J. Cano, C. Calafate, and P. Manzoni. A system for automatic notification and severity estimation of automotive accidents. *Mobile Computing, IEEE Transactions on*, PP(99):1–1, 2013.
- [25] Meng Guo, M.H. Ammar, and E.W. Zegura. V3: a vehicle-to-vehicle live video streaming architecture. In *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*, pages 171–180, 2005.
- [26] IEEE 802.11 WG. International Standard for Information Technology - Telecom. and Information exchange between systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC 8802-11:1999(E) IEEE Std. 802.11, 1999.
- [27] IEEE 802.11 WG. 802.11e IEEE Standard for Information technology- Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, 2005.
- [28] Hanzo II and Rahim Tafazolli. Admission control schemes for 802.11-based multi-hop mobile ad hoc networks: a survey. *Communications Surveys & Tutorials, IEEE*, 11(4):78–108, December 2009.

- [29] Jorge Hortelano, Juan-Carlos Cano, Carlos T. Calafate, and Pietro Manzoni. Testing applications in manet environments through emulation. *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, Article ID 406979, 20 pages, 2009. doi:10.1155/2009/406979.
- [30] K. Fall and K. Varadhan. ns notes and documents. The VINT Project. UC Berkeley, LBL, USC/ISI, and Xerox PARC, February 2000.
- [31] H. Koumaras, M. Kourtis, and Drakoulis Martakos. Benchmarking the encoding efficiency of h.265/hevc and h.264/avc. In *Future Network Mobile Summit (FutureNetw)*, 2012, pages 1–7, 2012.
- [32] Seoung-bum Lee, Gahng-Seop Ahn, Xiaowei Zhang, and Andrew T. Campbell. INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60:374–406, 2000.
- [33] Anders Lindgren and Elizabeth M. Belding-royer. Multi-path admission control for mobile ad hoc networks. In *in: Proceedings of Mobiquitous*, 2005.
- [34] FranciscoJ. Martinez, Manuel Fogue, Manuel Coll, Juan-Carlos Cano, CarlosT. Calafate, and Pietro Manzoni. Evaluating the impact of a novel warning message dissemination scheme for vanets using real city maps. In Mark Crovella, LauraMarie Feeney, Dan Rubenstein, and S.V. Raghavan, editors, *NETWORKING 2010*, volume 6091 of *Lecture Notes in Computer Science*, pages 265–276. Springer Berlin Heidelberg, 2010.
- [35] FranciscoJ. Martinez, Chai-Keong Toh, Juan-Carlos Cano, CarlosT. Calafate, and Pietro Manzoni. A street broadcast re-

- duction scheme (sbr) to mitigate the broadcast storm problem in vanets. *Wireless Personal Communications*, 56(3):559–572, 2011.
- [36] K. Mershad, H. Artail, and M. Gerla. We can deliver messages to far vehicles. *Intelligent Transportation Systems, IEEE Transactions on*, 13(3):1099–1115, Sept 2012.
- [37] Kulit Na Nakorn and Kultida Rojviboonchai. Deca-bewa: Density-aware reliable broadcasting protocol in vanets. *IE-ICE Transactions*, 96-B(5):1112–1121, 2013.
- [38] Anthony C. H. Ng, David Malone, and Douglas J. Leith. Experimental evaluation of TCP performance and fairness in an 802.11e test-bed. In *2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis*, pages 17–22, New York, NY, USA, 2005. ACM Press.
- [39] Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan Chen, and Jang-Ping Sheu. The broadcast storm problem in a mobile ad hoc network. In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom '99*, pages 151–162, New York, NY, USA, 1999. ACM.
- [40] T. Osafune, Lan Lin, and M. Lenardi. Multi-hop vehicular broadcast (mhvb). In *ITS Telecommunications Proceedings, 2006 6th International Conference on*, pages 757–760, 2006.
- [41] Pablo Piñol, Otoniel López, Miguel Martínez, José Oliver, and Manuel P. Malumbres. Modeling video streaming over vanets. In *Proceedings of the 7th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks, PM2HW2N '12*, pages 7–14, New York, NY, USA, 2012. ACM.

- [42] P. Pinol, A. Torres, O. Lopez, M. Martinez, and M.P. Malumbres. Evaluating hevc video delivery in vanet scenarios. In *Wireless Days (WD), 2013 IFIP*, pages 1–6, Nov 2013.
- [43] F.J. Ros, P.M. Ruiz, and I. Stojmenovic. Reliable and efficient broadcasting in vehicular ad hoc networks. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pages 1–5, 2009.
- [44] Julio A. Sanguesa, Manuel Fogue, Piedad Garrido, Francisco J. Martinez, Juan-Carlos Cano, Carlos T. Calafate, and Pietro Manzoni. An infrastructureless approach to estimate vehicular density in urban environments. *Sensors*, 13(2):2399–2418, 2013.
- [45] Patrick Seeling and Martin Reisslein. Video transport evaluation with h.264 video traces. *Communications Surveys Tutorials, IEEE*, 14(4):1142–1165, 2012.
- [46] Vasilios A. Siris and George Stamatakis. Optimal CWmin selection for achieving proportional fairness in multi-rate 802.11e WLANs: test-bed implementation and evaluation. In *1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, pages 41–48, New York, NY, USA, 2006. ACM Press.
- [47] F. Soldo, C. Casetti, C. Chiasserini, and P.A. Chaparro. Video streaming distribution in vanets. *Parallel and Distributed Systems, IEEE Transactions on*, 22(7):1085–1091, 2011.
- [48] Sven Wietholter and Christian Hoene. Design and Verification of an IEEE 802.11e EDCF Simulation Model in ns-2.26. Technical Report TKN-03-019, Telecommunication Networks Group, Technische Universitat Berlin, November 2003.

- [49] T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR). Request for Comments 3626, MANET Working Group, <http://www.ietf.org/rfc/rfc3626.txt>, October 2003. Work in progress.
- [50] Zhifeng Tao and S. Panwar. Throughput and delay analysis for the IEEE 802.11e enhanced distributed channel access. *IEEE Transactions on Communications*, 54(4):596–603, 2006.
- [51] Sergio M Tornell, Carlos T Calafate, Juan-Carlos Cano, and Pietro Manzoni. *Assessing the Effectiveness of DTN Techniques Under Realistic Urban Environments*. 2013.
- [52] Yu-Chee Tseng, Sze-Yao Ni, and En-Yu Shih. Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc network. *Computers, IEEE Transactions on*, 52(5):545 – 557, may 2003.
- [53] V. Jacobson, C. Leres, and S. McCanne. The libpcap packet capture library. Lawrence Berkeley Laboratory, Berkeley, CA. Available at <http://www.tcpdump.org>.
- [54] Celimuge Wu, Satoshi Ohzahata, and Toshihiko Kato. Vanet broadcast protocol based on fuzzy logic and lightweight retransmission mechanism. *IEICE Transactions*, 95-B(2):415–425, 2012.
- [55] T. Yan, W. Zhang, and G. Wang. A grid-based on-road localization system in vanet with linear error propagation, 2014.
- [56] Yaling Yang and Robin Kravets. Throughput guarantees for multi-priority traffic in ad hoc networks. *Ad Hoc Networks*, 5(2):228 – 253, 2007.