Giuseppe Bianchi
Andrey Lyakhov
Evgeny Khorov (Eds.)

LNCS 8072

Wireless Access Flexibility

First International Workshop, WiFlex 2013
Kaliningrad, Russia, September 2013
Proceedings

Springer

Lecture Notes in Computer Science 8072

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Giuseppe Bianchi Andrey Lyakhov
Evgeny Khorov (Eds.)

Wireless Access Flexibility

First International Workshop, WiFlex 2013
Kaliningrad, Russia, September 4-6, 2013
Proceedings

Springer

Volume Editors

Giuseppe Bianchi
Università degli Studi di Roma "Tor Vergata"
Dipartimento di Ingegneria Elettronica
Via del Politecnico, 1, 00133 Rome, Italy
E-mail: giuseppe.bianchi@uniroma2.it

Andrey Lyakhov
Evgeny Khorov
Russian Academy of Sciences
Institute for Information Transmission Problems
Bolshoy Karetny per., 19 b 1, 127994 Moscow, Russia
E-mail: {lyakhov, khorov}@iitp.ru

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-39804-9 e-ISBN 978-3-642-39805-6
DOI 10.1007/978-3-642-39805-6
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013943338

CR Subject Classification (1998): C.2, D.2, K.4

LNCS Sublibrary: SL 5 – Computer Communication Networks and Telecommunications

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Today's wireless devices and interfaces are mainly designed as monolithic, closed source systems. As such, they exhibit limited configuration and adaptation capabilities to the evolving needs of wireless operators and end customers. Innovation thus faces two major hurdles. On the one hand, new approaches and protocols must undergo a slow standardization process before being included in new products, even if modifications and extensions of existing standards are "marginal" and only related to tiny details. On the other hand even when technologies and solutions are provided via open and programmer-accessible frameworks (e.g., open source firmware or drivers), implementation complexity, platform dependency, and lack of modularity prevent all but core experts from successfully performing modifications in the wireless stack.

All this can be highly improved if existing wireless access standards based on closed architecture move toward new modular flexible architectures, opportunistically open for customization. Nowadays, many researchers from all over the world are involved in designing such architectures for the next generation wireless technologies (Google yields about 3,000,000 results for the keywords "flexible architecture of wireless networks").

Held in Kalinigrad, Russia, during September 4–6, 2013, the International Workshop on Wireless Access Flexibility (WiFlex 2013) was the first forum intended to address flexible wireless access architecture design that opens the door for innovative solutions significantly improving network performance even if they do not fully conform to current telecommunication standards.

It is our great pleasure to present the proceedings of WiFlex 2013. The contributions gathered in the book describe the latest results and novel research ideas. We would like to thank all authors and, of course, the Program Committee who did a wonderful job of critically reviewing the submissions and selecting the 13 strongest papers. Our gratitude goes to keynote speakers Prof. Mischa Dohler, Prof. Petri Mähönen, and Prof. Adam Wolisz, brilliant lectors and outstanding researchers, who gave insights into hot topics of wireless evolution. Also, we would like to express our gratitude to all the participants and local organizers, who helped to make WiFlex 2013 a very successful event, and to the Institute for Information Transmission Problems of the Russian Academy of Science (IITP RAS), the STRADO Scientific and Organizational Center, and the Russian Foundation for Basic Research (RFBR) for sponsorship.

September 2013

Giuseppe Bianchi
Andrey Lyakhov
Evgeny Khorov

Organization

WiFlex 2013 was organized by IITP RAS (Institute for Information Transmission Problems of the Russian Academy of Science) and supported by STRADO (Scientific and Organizational Center) and RFBR (Russian Foundation for Basic Research)

General Chairs

Giuseppe Bianchi	University of Rome Tor Vergata, Italy
Andrey Lyakhov	IITP RAS, Russia

Executive Chair

Evgeny Khorov	IITP RAS, Russia

Finance and Logistics Chair

Maxim Chuyashkin	STRADO, Russia

Visa Assistance

Elena Sidorova	IITP RAS, Russia

Program Committee

Jesus Alonso	Alvarion, Spain
Raffaele Bruno	IIT/CNR, Italy
Antonio Capone	Technical University of Milan/MobiMESH, Italy
Xavier Costa-Perez	NEC, Germany
Pierluigi Gallo	University of Palermo, Italy
Omer Gurewitz	Ben Gurion University, Israel
Evgeny Khorov	IITP RAS, Russia
Andreas Maeder	University of Wurzburg, Germany
David Malone	Hamilton Institute, Ireland
Vincenzo Mancuso	IMDEA, Spain
Enzo Mingozzi	University of Pisa, Italy
Marek Natkaniec	AGH, Poland

Paul Patras	Hamilton Institute, Ireland
Alexander Safonov	IITP RAS, Russia
Pablo Serrano	UC3M, Spain
Szymon Szott	AGH, Poland
Ilenia Tinnirello	University of Palermo, Italy
Guillaume Vivier	Sequans Communications, France

Keynotes

Machine-to-Machine in Smart Cities & Smart Grids: Vision, Technologies & Applications

Prof. Mischa Dohler, Kings College London, UK

The unprecedented communication paradigm of machine-to-machine (M2M), facilitating 24/7 ultra-reliable connectivity between a prior unseen number of automated devices, is currently gripping both industrial as well as academic communities. Whilst applications are diverse, the in-home market is of particular interest since undergoing a fundamental shift of machine-to-human communications towards fully automatized M2M. The aim of this keynote is thus to provide academic, technical and industrial insights into latest key aspects of wireless M2M networks, with particular application to the emerging smart city and smart grid verticals.

Notably, I will provide an introduction to the particularities of M2M systems. Architectural, technical and privacy requirements, and thus applicable technologies will be discussed. Notably, we will dwell on the capillary and cellular embodiments of M2M. The focus of capillary M2M, useful for real-time data gathering in homes, will be on IEEE (.15.4e) and IETF (6LoWPAN, ROLL, COAP) standards compliant low-power multihop networking designs; furthermore, low power Wifi will be dealt with and positioned into the eco-system of capillary M2M. The focus of cellular M2M will be on latest activities, status and trends in leading M2M standardization bodies with technical focus on ETSI M2M and 3GPP LTE-MTC.

Open technical challenges, along with the industrys vision on M2M and its shift of industries, will be discussed during the talk.

Looking in Rearview Mirror: Is there such thing as spectrum congestion?

Prof. Petri Mähönen, RWTH Aachen University, Germany

A number of recent studies, especially from different industry groups, predict an exponential increase of data traffic that could correspond to a 1000-fold increase in traffic within a decade. These estimations have led to the predictions of "spectrum congestion" or traffic crunch. The figures and discussion on required actions are very similar to ones deported during the last Internet boom a decade ago. In this talk we pose the fundamental question on what do we mean with the traffic load and follow with the question if we are experiencing the spectrum scarcity. We then explore various solutions to solve the possible problem with technological or other means. We specifically ask if Moores law can rescue once

more from the exponential traffic increase, or if something else is required. Finally we conclude on asking what is the role of access technology flexibility and software defined networking has on solving traffic problems.

How to Influence the Major Trends in Mobile Communication?

Prof. Adam Wolisz, TU Berlin, Germany

Accommodation of the predicted rapid growth of mobile data traffic is creating serious challenges.

In this talk I will separate the notion of mobile data service and the data transport needed for the service.

We argue that it is desirable and feasible to mitigate the traffic volume needed for achieving satisfactory services. In addition, we argue that traffic volume is not a proper measure reflecting the "transportation effort" in mobile communication. Various means to reduce this "transportation effort" and measures to quantify it will be presented. We conclude that more effort should be devoted to mitigation of the traffic growth as well as efficient traffic organization in contrast to accommodating the traffic explosion by purely blowing up the communication capacity.

In this talk the above issues will be discussed from the technology point of view. In addition we will also present some thoughts about possible new charging models able to encourage the desired trends.

Table of Contents

4G and beyond

Delay and Loss Due to Uplink Packet Scheduling in LTE Network 1
Alia Asheralieva, Kaushik Mahata, and Jamil Y. Khan

Distributed Adaptive Interference Control in 4G Small Cell Networks ... 13
Evgeni Bikov and Stanislav Elizarov

Smart Cities Software: Customized Messages for Mobile Subscribers 25
Manfred Sneps-Sneppe and Dmitry Namiot

Local Area Networks

802.11 Buffers: When Bigger Is Not Better? 37
David Malone, Hanghang Qi, Dmitri Botvich, and Paul Patras

On the Capacity of a PPM UWB Multiple-Access System with a Single User Noncoherent Reception 49
Dmitry Osipov, Alexey Frolov, and Victor Zyablov

Detecting Transmission Power Misbehaviour in Wi-Fi Networks 58
Szymon Szott, Marek Sikora, Marek Natkaniec, and Krzysztof Loziak

P-Persistent Queue Management to Overcome Channel Failures in IEEE 802.11 Networks for Real-Time Multimedia Streaming 69
Andrey Guschin, Evgeny Khorov, Anton Kiryanov, Andrey Lyakhov, and Alexander Safonov

Multi-hop Networks

Supporting a Pseudo-TDMA Access Scheme in Mesh Wireless Networks 80
Ilenia Tinnirello and Pierluigi Gallo

Dynamic Resource Allocation for MCCA-Based Streaming in Wi-Fi Mesh Networks 93
Evgeny Khorov, Artem Krasilov, Andrey Lyakhov, and Dmitry Ostrovsky

Distributed Clock Synchronization Algorithm for Wide-Range TDMA Ad Hoc Networks 112
Dmitry Doronin and Denis Fakhriev

Joint Integrated Spectrum Handoff Management and Routing in CR-MANETs: An Analytical Modeling 125
S. Nejatian, S.K. Syed-Yusof, N.M. Abdul Latiff, V. Asadpour, and N. Fisal

Sensor Networks

Dimensioning Self-sufficient Networks of Energy Harvesting Embedded Devices 138
Nicola Bui and Michele Rossi

Experiment Design for Parameter Estimation in Sensing Models 151
Vladimir V. Shakhov

Author Index 159

Delay and Loss Due to Uplink Packet Scheduling in LTE Network

Alia Asheralieva, Kaushik Mahata, and Jamil Y. Khan

University of Newcastle, School of Electrical Engineering and Computer Science,
Callaghan NSW 2308, Australia
alia.asheralieva@uon.edu.au,
{Kaushik.Mahata,Jamil.Khan}@newcastle.edu.au

Abstract. In this paper we describe the packet scheduling process and investigate the reasons limiting the medium access control (MAC) layer capacity of the 3rd Generation Partnership Project Long Term Evolution (3GPP LTE) network. We show that although a scheduling process allows to assign dedicated channels to the users based on their quality of service (QoS) requirements, it introduces the additional delay in the uplink channel. We also show that the scheduling delay may increase significantly if some certain parameters of the system are not set appropriately, and suggest alternative approaches to reduce the scheduling delay in LTE network. Obtained analytical expressions of the packet scheduling delay and loss have been verified using simulation model developed in OPNET platform. Results of this work can be used for resource allocation, packet scheduling and network planning to establish the upper bounds on delay and loss for the users with strict QoS requirements.

Keywords: 3GPP LTE, packet scheduling, performance evaluation.

1 Introduction

Today a 3rd Generation Partnership Project Long Term Evolution (3GPP LTE) is considered to be the main standard for deployment in future wireless networks. In the downlink the LTE system uses Orthogonal Frequency-Division Multiple Access (OFDMA) due to its high spectral efficiency and robustness against interference. In the uplink a Single Carrier Frequency Division Multiple Access (SC-FDMA) is utilized because of its lower Peak-to-Average Power Ratio (PAPR) compared to traditional OFDM [1]. In LTE, available transmission resources are distributed among the users by the medium access control (MAC) schedulers in enhanced NodeBs (eNBs). LTE standardizes control signaling and a general framework on physical and MAC layer. An exact algorithm for resource allocation is not specified: depending on the implementation, it can be based on the queuing delay, instantaneous channel conditions, fairness, etc. Thus, the scheduling process allows assigning dedicated channels to the users based on their quality of service (QoS) requirements [2]. However, it introduces the additional delay in the uplink channel. In some cases (for instance, for real-time applications) such impact on the end-to-end service performance of the

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 1–12, 2013.

network cannot be neglected. Therefore, it is very important to analyze the delay due to packet scheduling in LTE system.

The scheduling performance for different types of users has been studied in many papers. Various multiuser scheduling strategies in the context of OFDMA downlink have been described in [3-7]. The uplink capacity of LTE system has been investigated in [8] and [9]. Although these works provide a closer look on the capacity and coverage of LTE network depending on channel conditions, resulting end-to-end performance of wireless communication systems is evaluated only by means of simulations, and no analytical verification of obtained results is conducted. The average values of various delay components including delay due to packet scheduling have been given in [10]. However, no proper mathematical analysis confirming the delay values have been presented.

In this paper we describe the packet scheduling process and investigate the reasons limiting the MAC layer capacity of LTE network. Based on LTE standard specifications, we provide a complete analysis of the delay and loss due to packet scheduling in LTE system. Obtained analytical expressions of the packet scheduling delay and loss are applicable to any resource allocation algorithm, and can be used in the analysis of the end-to-end packet delay for the users of LTE network. The rest of the paper is organized as follows. In Section 2 we provide some background information on the design issues of LTE system related to the packet scheduling process, and derive the expressions for packet scheduling delay and loss estimation. In Section 3 we present the simulation framework conducted in OPNET environment [15] to validate the analytical expressions of the scheduling delay and loss, and discuss the possible ways to reduce the scheduling delay and loss in the network. The conclusions are drawn in Section 4.

2 Packet Scheduling Process

2.1 The General Uplink Packet Scheduling Procedure

In LTE, the downlink transmission scheme is based on conventional OFDM. In an OFDM system the available spectrum is divided into multiple subcarriers, which are modulated independently by a low rate date stream. The main advantages of OFDM are its robustness against multipath fading and efficient receiver architecture. Besides, OFDMA-based channel access supports multiple users on the available bandwidth, because within one transmission time interval (TTI) subcarriers can be allocated to different users. The uplink transmission scheme is based on SC-FDMA, which has better PAPR properties then OFDMA-based signals [2, 11]. The basic radio resource unit in LTE is called a resource block (RB). In frequency domain one RB consists of 12 subcarries with a constant subcarrier spacing Δf = 15 kHz. In time domain it has length equal 1TTI with duration T_s = 1 ms. The number of RBs, denoted N_{RB} depends on the channel bandwidth B. The capacity of one RB depends on the Modulation and Coding Scheme (MCS) which determines the bit rate [12, 13].

In LTE, resources are allocated to user equipments (UEs) for uplink and downlink data transmission in terms of RBs. Thus, one UE can be allocated only the integer number of RBs in frequency domain, and these RBs do not have to be adjacent to each other. Resource allocation (scheduling) is carried by the MAC layer packet scheduler in the eNB both for uplink and downlink transmissions. Resource allocation (scheduling) is usually performed periodically within a fixed time interval, called scheduling period T_{sc}. Depending on its implementation, the scheduler can allocate resources based on the quality of service (QoS) requirements, instantaneous channel conditions, fairness, etc. Besides, the scheduler has to ensure that Hybrid Automatic Repeat Request (HARQ) retransmissions are performed on a timely basis (in LTE system a packet retransmission should be send in exactly 8 ms after receiving a Negative Acknowledgement message) [2, 12, 13].

After resource allocation, the user data are carried by the Physical Uplink Shared Channel (PUSCH) in uplink direction and the Physical Downlink Shared Channel (PDSCH) in downlink direction. The scheduling decisions are carried by the Physical Uplink Control Channel (PUCCH) and Physical Downlink Control Channel (PDCCH) in uplink and downlink directions, respectively [14]. The general scheduling procedure shown on Figure 1 consists of the following steps [10, 13]:

1. First SR-SG exchange step:
   (a) UE generates initial scheduling request (SR) without any scheduling information, and starts the scheduling timer with a timeout T_{sc}.
   (b) If the eNB receives the SR, eNB transmits first scheduling grant (SG) to notify the UE that it is waiting for the buffer status report (BSR) data with scheduling information.
   (c) If the UE does not receive the SG before the scheduling timer expires, it repeats step 1(a).
2. Second SR-SG exchange step:
   (a) UE sends secondary SR with BSR MAC Control Element, and starts the BSR retransmission timer with a timeout T_{BSR}. The BSR MAC Control Element carries the information about the amount of uplink data waiting to be transmitted by UE.
   (b) If the eNB receives the second SR, it allocates the resources to UE and reports about allocation by generating secondary SG. The amount of resources allocated to UE depends on the received BSR information.
   (c) If the UE does not receive the SG before the BSR retransmission timer expires, it repeats step 2(a).
3. Third step:
   (d) After the secondary SG is received, a UE transmits the uplink data.

According to LTE standard, a UE is allowed to transmit at most N_{max} SRs per packet, after which the packet is dropped.

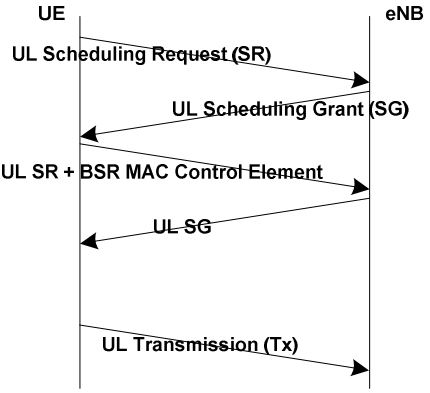


Fig. 1. The general uplink scheduling procedure

2.2 PUCCH and PDCCH Configuration

According to LTE standard, PUCCH carries the following uplink control information: HARQ Acknowledgement (ACK) and Negative Acknowledgement (NACK) information related to data packets received in downlink; scheduling requests (SRs) for packet scheduling; Channel Quality Indicator (CQI) reports for Adaptive Modulation and Coding (AMC); pre-coding matrix information (PCI) and rank indication for Multiple-Input-Multiple-Output (MIMO) [13]. Depending on the type of information, the LTE specifies different PUCCH formats. In particular, SRs are carried by the PUCCH format 1/1a/1b messages. According to the standard, a determined number of RBs N_{RB_1} are reserved for PUCCH format 1/1a/1b transmissions, and each reserved RB is further divided into a number of resource indexes (RIs). One RI is allocated for one PUCCH message. The number of RIs in a RB depends on message format. For format 1/1a/1b messages the direct mapping between PUCCH cyclic shifts and RIs cannot be used because of the block-spreading operation. For this purpose, PUCCH channelization is used to provide a number of parallel sub-channels with adjustable orthogonal properties, which are configured by means of the special system parameter δ (allowed values are δ = 1, 2 or 3). It is periodically broadcasted in the network to maintain the orthogonality of the sub-channels. The number of parallel orthogonal sub-channels (or RIs) per a RB allocated for PUCCH format 1/1a/1b messages N_{RI_1} can be calculated from [10, 11 - 14]:

$$N_{RI\_1} = \frac{12N_{RS}}{\delta} \tag{1}$$

where N_{RS} is the number of reference signals on PUCCH format 1/1a/1b (N_{RS} = 3 for normal CP, N_{RS} = 2 for extended CP) [19]. Then, the total number of RIs allocated for a format 1/1a/1b messages N_{PUCCH_1} is equal [10]:

$$N_{PUCCH\_1} = N_{RB\_1} N_{RI\_1} = \frac{12 N_{RB\_1} N_{RS}}{\delta} \tag{2}$$

For instance, for normal CP with N_{RB_1} = 1 RB the allowed numbers of allocated channels are N_{PUCCH_1} = 12, 18 or 36 channels.

PDCCH is primarily used to carry the Downlink Control Information (DCI), such as number of OFDM symbols reserved within each time slot for PDCCH signals; scheduling information (downlink assignments and uplink SGs); HARQ and MCS. According to LTE specifications, PDCCH occupies the first 1, 2, or 3 OFDM symbols in a time slot extending over the entire system bandwidth. PDCCH is constructed from Control Channel Elements (CCEs). The number of CCEs N_{CCE} indicates the capacity of PDCCH: each UE generating a DCI message within a considered time slot is assigned a CCE (N_{CCE} for 5, 10 and 20 MHz bandwidth is given in Table 1) [14].

Table 1. The Number of CCEs for 5, 10, 20 MHz Bandwidth [14]

The number of CCEs per time slot, N_{CCE}	**Bandwidth, B (MHz)**		
	5	**10**	**20**
1 PDCCH symbol per time slot	3	8	17
2 PDCCH symbols per time slot	12	25	50
3 PDCCH symbols per time slot	20	41	84

2.3 Delay and Loss Due to Uplink Packet Scheduling

Delay and loss associated with the transmission of an SR or an SG depends on how fast the UE and eNB are able to get through the first and second steps. In particular, the delay associated with the transmission of one SR consists of the transmission, buffering, propagation and processing delays in the uplink direction. Similarly, the delay associated with the transmission of one SG comprises the transmission, buffering, propagation and processing delays in the downlink direction. For further analysis we assume, that the processing time for an SR/SG is equal to one TTI with duration T_s = 1 ms. The transmission, buffering and propagation delays for an SR/SG are in order of 1 μs, which is very small compared to T_s. Thus, the delay associated with the transmission of an SR/SG can be accurately estimated by T_s.

Now we are ready to state the main results of this paper, that is, to estimate the mean packet delay and loss due to uplink packet scheduling. In our analysis we consider the basic LTE network comprising one eNB and n active UEs sending SRs to eNB independently in a random fashion. The time interval between two consecutive SRs generated by a particular UE is fixed and equal T_{sc}. The eNB responds to each successfully received SR by sending an SG to UE.

Theorem 1: The probability of success for an SR-SG exchange attempt between the UE and the eNB in such network can be estimated by

$$p_s = \begin{cases} 1, \text{ if } n \le \dfrac{T_{SR}}{T_s} \times \min\{N_{PUCCH\_1}, N_{CCE}\} \\ \dfrac{T_{SR}}{T_s} \times \dfrac{\min\{N_{PUCCH\_1}, N_{CCE}\}}{n}, \text{ otherwise} \end{cases} \tag{3}$$

Proof: See Appendix A.

Lemma 1: If the probability of success for a particular SR-SG exchange attempt between the UE and the eNB is given by p_s, then the probability that a packet is lost in the scheduling process is

$$P_{sc} = (1-p_s)^{N_{\max}} + N_{\max} p_s (1-p_s)^{N_{\max}-1} \tag{4}$$

Proof: A packet is lost in the scheduling process if there is at most one success in N_{max} consecutive SR/SG scheduling transmission attempts, probability of which is given by (6) using the well-known results on repeated trials.

Theorem 2: The packets which are successfully scheduled experience an average uplink packet scheduling delay

$$\begin{aligned} &T_{PS} = 4T_s + \frac{1}{A}(BT_{sc} + CT_{BSR}), \\ &\text{where: } A = 1 - q_s^{N_{\max}-2}(1 - N_{\max} p_s), \quad B = q_s\left[1 - q_s^{N_{\max}-3}\left\{2N_{\max} - 3 + 2p_s + (N_{\max}-2)p_s^2\right\}\right], \\ &C = \frac{2 - q_s^{N_{\max}-2}\left\{2 + 2(N_{\max}-2)p_s + (N_{\max}-2)(N_{\max}-3)p_s^2\right\}}{2p_s}, \quad q_s = 1 - p_s. \end{aligned} \tag{5}$$

Proof: See Appendix B.

3 Simulation Framework

In this section we provide the numerical validation of the derived analytical expressions for the packet scheduling delay and loss given by (3) - (5) by comparing the values of the mean packet scheduling delay and loss obtained analytically with the results gathered in simulations. The simulation model of the network (show on Figure 2) has been developed according to requirements of LTE standard [13, 14] using OPNET platform [15]. The users generate a mixed traffic comprising of voice, video and data applications in proportion 1:1:2, respectively. The user traffic is simulated in accordance with the requirement stated in [16]. Voice traffic is generated by using the G.723.1 (12.2 Kbps) codec with a voice payload size 40 bytes and a voice payload interval 30 ms. Each voice user might be either in active (talk-spurts period) or inactive (silent period) state. The durations of the talk-spurts and silent periods are exponentially distributed with 0.65s and 0.352s means, respectively. Video services are simulated using a high resolution video model with a constant frame size equal 6250 bytes and exponentially distributed frame inter-arrival intervals (with mean equal

0.5s). Data users in simulations are HTTP1.1 users generating pages or images with exponential page inter-arrival intervals (mean equal 60s). It is assumed that one page consists of one object, whereas one image consists of five objects. The object size is constant and equal 1000 bytes. Simulations have been carried for the networks with B = 5 MHz and B = 10 MHz bandwidth. The other parameters necessary to estimate the packet scheduling delay and loss are listed in Table 2.

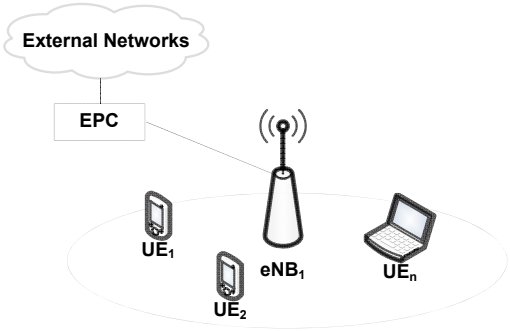


Fig. 2. The simulation model of the network

Table 2. Common Simulation Parameters

Parameter	Value
Bandwidth, B	5MHz, 10MHz
Cyclic Prefix Type	Normal
PDCCH symbols per subframe	3
PUCCH Reserved Size for format 1 messages, N_{RB_1}	1 RB
Cyclic shift, δ	3
TTI duration, T_s	1 ms
Scheduling Timer timeout, T_{sc}	1 ms
BSR Retransmission Timer timeout, T_{BSR}	256 ms
Maximal number of SRs per packet, N_{max}	10

Figures 3, 4 show the mean packet scheduling delay and loss estimated analytically and obtained using simulations (experimentally) for the networks with B = 5 MHz and B = 10 MHz bandwidth, respectively. Results on Figure 3 show that in the network with B = 5 MHz the scheduling delay and loss start increasing when the number of users n > 20 users. In the netowork with B = 10 MHz the scheduling delay and loss stat growing when n > 35 users. Obtained results correspond to expressions (3) - (5). In particular, for $n \le T_{SR}/T_s \times \min\{N_{PUCCH\_1}, N_{CCE}\}$ UEs the probability of the success for SR-SG exchange p_s = 1 and the values of the scheduling delay and loss are equal $T_{PS} = 4T_s$ = 4 ms and P_{PS} = 0, respectively. Starting from the point $n = T_{SR}/T_s \times \min\{N_{PUCCH\_1}, N_{CCE}\}$ UEs the probability of success for SR-SG exchange p_s start to decrease, and consequently the delay and loss due to packet scheduling T_{PS} and P_{PS} start growing.

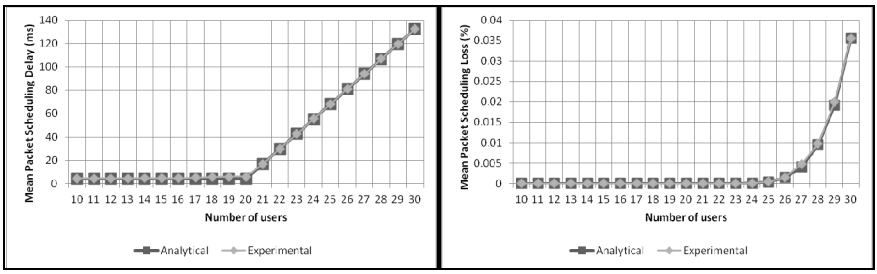


Fig. 3. Analytical and experimental values of the scheduling delay and loss for B = 5 MHz

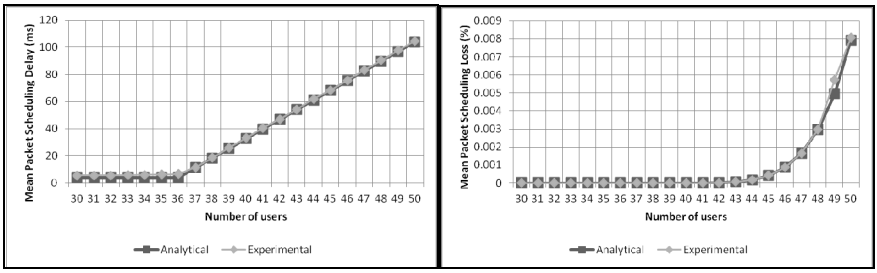


Fig. 4. Analytical and experimental values of the scheduling delay and loss for B = 10 MHz

To finalize this section, we note that the scheduling delay and loss cannot be neglected if the number of active users of the network exceeds the number of available PUCCH parallel subchannels N_{PUCCH_1} or the number of PDCCH control channel elements N_{CCE}. However, as it follows from expressions established by the Theorems 1 and 2, there exist a number of ways to reduce the delay due to packet scheduling. The easiest and most obvious way is to increase N_{PUCCH_1} (by increasing the number of RBs reserved for PUCCH) and N_{CCE} (by increasing the number of PDCCH symbols per subframe). In this case, however, the number of RBs reserved for data will decrease, which may eventually lead to overall QoS degradation.

An alternative strategy to reduce the expected delay and loss in the network would be to decrease the BSR retransmission timer timeout T_{BSR} (in LTE, T_{sc} is in orders of T_s, and therefore its impact on the scheduling delay is neglectably small). To validate this claim, a number of simulations have been conducted in the network with B = 5MHz bandwidth and different BSR timeouts T_{BSR} = 2520ms and T_{BSR} = 320ms. Figure 5 illustrate the packet end-to-end delay and loss for the network users. Obtained results confirm our expectations – delay and loss decreases in times, which can be explained by faster network reaction on resource re-allocation during retransmissions. Therefore, we recommend to set the BSR retransmission timer timeout T_{BSR} as small as possible to minimize the scheduling delay (which is especially important in case of delay-sensitive applications, such as voice or video). Note, that in order to allow the serving eNB to receive and send the BSR MAC Control Element, as well as to re-allocate the resources, the BSR retransmission timer should not be less than the maximal packet round trip time. Thus, the maximal packet round trip time will serve as the lower bound for BSR retransmission timer timeout T_{BSR} in LTE system.

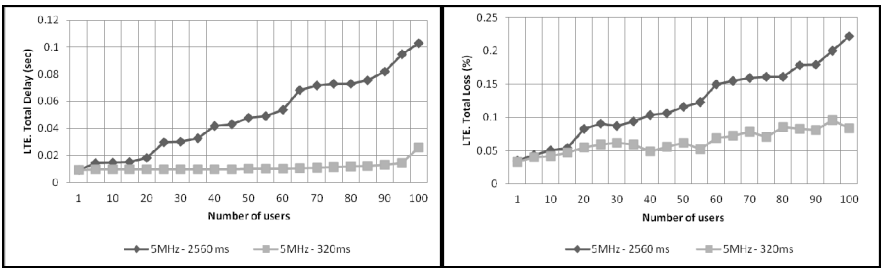


Fig. 5. Packet end-to-end delay and loss in the network with B = 5 MHz

4 Conclusions

This paper presents the analytical method to estimate the delay and loss due to packet scheduling in LTE network. The expressions derived in the paper can be used for resource allocation, packet scheduling and network planning. Simulation results have shown that the theoretically obtained values of the scheduling delay and loss closely follow the actual values. It has also been found, that the values of the scheduling delay and loss grow rapidly when the number of users in the network exceeds some certain bounds. To prevent this multiplicative growth of delay, a couple of alternative strategies have been proposed and discussed in the paper.

Appendix A: Proof of Theorem 1

In LTE standard, SRs are carried via PUCCH in format 1/1a/1b messages, SGs are carried via PDCCH in DCI messages (see description of PUCCH and PDCCH provided in Section II.D). This means that each TTI there are exactly N_{PUCCH_1} parallel sub-channels available for all SRs generated by UEs to eNB in uplink direction, and NCCE control channel elements reserved for all SGs generated by eNB to UEs in downlink direction. A particular SR-SG exchange between the UE and the eNB is successful if both SR and SG are received successfully by the eNB and the UE, respectively. Let p_s^{SR} be the probability of a successful reception of an SR from UE by eNB, and p_s^{SG} be the probability of a successful reception of an SG from eNB by UE. Then, the probability of a successful SR-SG exchange between the UE and the eNB ps is equal to the product of both of these probabilities

$$p_s = p_s^{SR} p_s^{SG} \tag{6}$$

If N_{SR} is the number of SRs generated within one TTI, then it is readily verified that

$$p_s^{SR} = \begin{cases} 1, \text{if } N_{SR} \le N_{PUCCH\_1} \\ \dfrac{N_{PUCCH\_1}}{N_{SR}}, \text{otherwise} \end{cases} \tag{7}$$

or:

$$p_s^{SR} = \frac{\min\{N_{SR}, N_{PUCCH\_1}\}}{N_{SR}} \tag{8}$$

Let N_{SG} be the number of SGs generated within one TTI. The eNB can receive at most N_{PUCCH_1} SRs from the UEs, and can respond to at most N_{CCE} users. This means that

$$N_{SG} = \min\{N_{SR}, N_{PUCCH\_1}\} \tag{9}$$

$$p_s^{SG} = \begin{cases} 1, \text{if } N_{SG} \le N_{CCE} \\ \dfrac{N_{CCE}}{N_{SG}}, \text{otherwise} \end{cases} \tag{10}$$

or:

$$p_s^{SG} = \frac{\min\{N_{SG}, N_{CCE}\}}{N_{SG}} \tag{11}$$

Combining (8), (9) and (11) we get:

$$p_s = \frac{\min\{N_{SR}, N_{PUCCH\_1}, N_{CCE}\}}{N_{SR}} \tag{12}$$

or:

$$p_s = \begin{cases} 1, \text{if } N_{SR} \le \min\{N_{PUCCH\_1}, N_{CCE}\} \\ \dfrac{\min\{N_{PUCCH\_1}, N_{CCE}\}}{N_{SR}}, \text{otherwise} \end{cases} \tag{13}$$

In the considered scenario the eNB serves n active users, sending SRs to eNB periodically within the interval T_{sc}. Let p_{SR} be the probability that a single UE will generate SR within a particular TTI. It readily follows that

$$p_{SR} = \frac{T_s}{T_{sc}} \tag{14}$$

Taking into account that UEs generate the SR randomly and independently, the probability that k of the n UEs will generate the SRs within any particular TTI denoted via $P_{SR}(k)$ is binomially distributed, i.e.:

$$P_{SR}(k) = \binom{n}{k} p_{SR}^k (1 - p_{SR})^{n-k} \tag{15}$$

where:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{16}$$

Then, the mean number of SRs generated in the network is equal

$$N_{SR} = np_{SR} \tag{17}$$

Appendix B: Proof of Theorem 2

Suppose the first SG is received after n_1 SR attempts, the second SG is received after n_1 SR attempts. Note that n_1 and n_2 are random variables. It is straightforward to check that

$$\Pr\{n_1 = i, n_2 = j\} = p_s^2 q_s^{i-1} q_s^{j-1} \tag{18}$$

Let S denote that event that the scheduling is done successfully. Hence

$$\Pr\{S\} = \text{prob}\{n_1 + n_2 \le N_{\max}\} = \sum_{i=1}^{N_{\max}-1} \sum_{j=1}^{N_{\max}-i-1} p_s^2 q_s^{i-1} q_s^{j-1} \tag{19}$$

Using Bayes' theorem

$$\Pr\{n_1 = i, n_2 = j \mid S\} = \frac{p_s^2 q_s^{i-1} q_s^{j-1}}{\sum_{i=1}^{N_{\max}-1} \sum_{j=1}^{N_{\max}-i-1} p_s^2 q_s^{i-1} q_s^{j-1}} \tag{20}$$

Let the time needed to obtain the first SG be t_1, and the time needed to obtain the second SG be t_2. Then

$$t_1 = 2T_s + (n_1 - 1)T_{sc} \tag{21}$$

because, the waiting time for n_1-1 unsuccessful attempts is $(n_1-1)T_{sc}$, while the total time needed for successful transmission of SR and SG is $2T_s$ (a T_s to transmit SR and another T_s to transmit SG). Similarly,

$$t_2 = 2T_s + (n_2 - 1)T_{BSR} \tag{22}$$

as the waiting time for n_2-1 unsuccessful attempts is $(n_2-1)T_{BSR}$, while the time needed for successful transmission is $2T_s$. Hence the total scheduling delay experienced by the packet is given by

$$t_1 + t_2 = 4T_s + (n_1 - 1)T_{sc} + (n_2 - 1)T_{BSR} \tag{23}$$

Since n_1 and n_2 are random variables, so are t_1 and t_2. Now

$$\begin{aligned} T_{PS} &= E\{t_1 + t_2 \mid S\} = \sum_{i=1}^{N_{\max}-1} \sum_{j=1}^{N_{\max}-i-1} \Pr\{n_1 = i, n_2 = j \mid S\}\left[4T_s + (i-1)T_{sc} + (j-1)T_{BSR}\right] \\ &= \frac{\sum_{i=1}^{N_{\max}-1} \sum_{j=1}^{N_{\max}-i-1} p_s^2 q_s^{i-1} q_s^{j-1}\left[4T_s + (i-1)T_{sc} + (j-1)T_{BSR}\right]}{\sum_{i=1}^{N_{\max}-1} \sum_{j=1}^{N_{\max}-i-1} p_s^2 q_s^{i-1} q_s^{j-1}} \end{aligned} \tag{24}$$

After some modifications to (24) we get the expression given by (5).

References

1. Berardinelli, G., et al.: OFDMA vs. SC-FDMA: Performance comparison in local area IMT-A Scenarios. IEEE Trans. Wireless Commun. 15(5), 64–72 (2008)
2. Wong, C.Y., et al.: Multiuser OFDM with adaptive subcarrier, bit and power allocation. IEEE J. Select. Areas Commun. 17(10), 1747–1758 (1999)
3. Kivanc, D., Li, G., Liu, H.: Computationally efficient bandwidth allocation and power control for OFDMA. IEEE Trans. Wireless Commun. 2(6), 1150–1158 (2003)
4. Ergen, M., Coleri, S., Varaiya, P.: QoS aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless systems. IEEE Trans. Broadcasting 49(4), 362–370 (2003)
5. Song, G., Lee, Y.: Cross-layer optimization for OFDM wireless networks–part I-II. IEEE Trans. Wireless Commun. 4(2), 614–634 (2005)
6. Yin, H., Liu, H.: An efficient multiuser loading algorithm for OFDM based broadband wireless systems. In: Proc. IEEE GLOBECOM (2000)
7. Boussif, M., et al.: Interference Based Power Control Performance in LTE Uplink. In: Proc. IEEE ISWCS, pp. 698–702 (2008)
8. Wang, H., Jiang, D.: Performance Comparison of Control-less Scheduling Policies for VoIP in LTE UL. In: Proc. IEEE WCNC, pp. 2497–2501 (2008)
9. Holma, H., Toskala, A.: LTE for UMTS: Evolution to LTE-Advanced, p. 576. John Wiley and Sons (2011)
10. Larmo, A., et al.: The LTE link-layer design. IEEE Comm. Mag. 47(4), 52–59 (2009)
11. 3GPP TS 25.892. Feasibility Study for Orthogonal Frequency Division Multiplexing (OFDM) for UTRAN enhancement (Release 6)
12. Kela, P., et al.: Dynamic Packet Scheduling Performance in UTRA Long Term Evolution Downlink. In: Proc. IEEE ISWCS (2008)
13. 3GPP TS 36.321. E-UTRA; MAC protocol specification (Release 8)
14. 3GPP TS 36.211. Physical Channels and Modulation (Release 8)
15. OPNET official site: `http://www.opnet.com`
16. 3GPP TS 26.114. Multimedia Telephony; Media handling and interaction (2007)

Distributed Adaptive Interference Control in 4G Small Cell Networks

Evgeni Bikov and Stanislav Elizarov

Telum, Trubnaya ulitsa 23, 127051 Moscow, Russia
{bikov,elizarov}@telum.ru

Abstract. Recently small cell networks have gained significant attention. Providing considerable capacity gain such systems bring heavy resource competition as well as mutual interference between multiple cells. In this paper we propose a distributed solution for interference control tailored for the use within small cell networks. We demonstrate the effectiveness of path loss difference for characterizing the need to divide resources between small cells in a non-uniform deployment scenario. We show the possibility of flexible transition between different interference control policies and demonstrate the presence of an optimum parameter values for which the performance of the users experiencing poor channel conditions is growing without compromising overall network throughput. We compare our solution with the basic resource control approaches and discuss the possibilities to enhance it in order to achieve fairness of resource distribution accurately.

Keywords: 4G mobile communication, cellular networks, multiple access interference, telecommunication control, wireless communication.

1 Introduction

The growing demand for wireless cellular technologies, capable to support data-intensive services, led to emergence of the 4G networks. One of the key requirements to this systems is the maximization of a frequency reuse factor. The use of overlapping frequency bands inevitably leads to an inter-cell interference. This serves as the reason of intensive scientific activity in the field of interference control. In the industrial plane, these efforts were largely embodied in the algorithms of static pre-planned frequency division or reactive (as opposed to proactive) management of the available spectrum.

The emergence of the concept of small cells became one of the logical steps on the way of increasing the total system capacity. In this approach the use of the large number of low-power devices (small cells) is meant for increasing spatial frequency reuse. The approach fundamentally changes the familiar face of cellular networks. One of the key features of this rapidly developing concept is that small cells are quite often being deployed with little control from the operator. As a result, a network of small cells can essentially differ in structure comparing with a network of macrocells. Also, devices switching on and off can cause uncontrolled frequent changes in the structure of small cell network.

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 13–24, 2013.

The concept of small cells challenges many time-honored assumptions about the structure of cellular networks [1]. As a result, one cannot apply habitual interference control approaches as they are mostly based on the assumptions of uniform hexagonal deployment and centralized management of base stations. Analysts expects the small cell market to experience significant growth over the next few years, reaching 91.9 million small cells by 2016 [2]. Together with the growth of the number of devices the quantity of service overhead increases. The use of centralized solutions become less favorable in such circumstances. At the same time distributed algorithms for resource management become a matter of great urgency.

Most of the modern approaches to interference control use inherited out-of-date notion about network topology and network management methods. Absolute majority of authors (e.g. [3–7]) use the scenario where base stations are evenly placed on a hexagonal grid to validate their algorithms. This makes it impossible to catch any specifics of non-uniform (up to three-dimensional non-uniform [6]) deployment scenario of small cells. Chung and Tao [4, 5] propose to construct a so called *interference graph* of cellular network users with subsequent marking out of strongly connected components and interference coordination within them. The suggested solutions for interference control within a network of base stations are often difficult to scale (e.g. see [8]) and usually require a centralized controller, which further complicates the use within small cell scenarios.

Quite often it turns out that the proposed approach can not be implemented in practice for several reasons. For operation of an algorithm one might demand the knowledge of a global condition of all network [3], or e.g. the solution of essentially complex computational problems [4, 5] can be required. Often in papers it is offered to use an additional information, for example, a geographical position [5] of users or an assessment of interference levels [4]. In many studies the ideal knowledge of SINR [9] is supposed, or this value is estimated at the stage of resource allocation [5] what is rather rough approach.

In this paper we propose a practical distributed adaptive algorithm for interference control tailored for the use within a network of LTE small cells. In particular, we construct an interference graph and allocate resources in a distributed manner. Also we present an algorithm extension which enhances the basic solution in terms of throughput and fairness of resource distribution. The gains are achieved through the interaction with a scheduler on base station at the stage of resource allocation.

This paper is organized as follows. In Section 2 we formulate the statement of the problem and describe the specifics of interference control within a network of small cells. Section 3 presents the developed algorithms. In section 4 we show simulation results and discuss the effectiveness of the proposed solutions. We conclude the paper with the discussion of possible improvements of the proposed methods in Section 5.

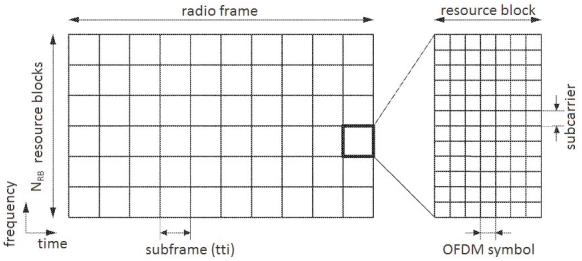


Fig. 1. LTE frame structure

2 System Description

We consider LTE downlink cellular system with N base stations, each serving m users. Each station can establish connection with any other station by means of backhaul link ("X2 interface" in 3GPP terms). It can be used then for control signaling. LTE supports localized OFDM where adjacent symbols and subcarrier are scheduled. The decision on resource usage is taken on 1 ms basis. Hereinafter this period of time is referred to TTI (Transmission Time Interval). Resources are grouped into two dimensional Resources Blocks (see Fig. 1). A Resource Block (RB) is the minimum unit a scheduler can allocate. Thus, N_{RB} resource blocks are available for each base station at any given time.

There are two common interference control algorithm in cellular networks: 1-Reuse and N-Reuse. They imply a division of resource blocks at each base station into two classes - captured and uncaptured. If the resource block for a given base station belongs to the captured class, then the signal can be transmitted to the user with maximum power P_0, otherwise output power is limited. Algorithm 1-Reuse implies that base station transmits in the whole system bandwidth. N-Reuse algorithm generally treats only N-th part of the resource blocks as captured so that neighbouring cells do not use the same set of frequencies.

To evaluate the performance of the interference control algorithms, we estimate the throughput of user i in the resource block k as follows. At first we calculate $SINR_k^{(i)}$ value by the formula (1):

$$SINR_k^{(i)} = \frac{P_k^{(i)} Pl_i^{(i)}}{\sum_u P_k^{(u)} Pl_u^{(i)} + T}, \tag{1}$$

where $P_k^{(i)}$ is the signal power in resource block k for the station that serves user i, $Pl_u^{(i)}$ is the signal attenuation between user i and the base station that serves user u and T denotes to thermal noise power. Next, $SINR_k^{(i)}$ value is mapped to the spectral efficiency according to (2) as it recommended by the 3GPP [10].

$$SINR_k^{(i)} \Rightarrow CQI_k^{(i)} \Rightarrow MCS_k^{(i)} \Rightarrow \text{spectral efficiency} \tag{2}$$

The evaluation setup also follows closely the suggestions given by the 3GPP LTE for small cell scenario [10].

3 Proposed Algorithms

3.1 Basic Proposed Scheme

Step I: Determine interference graph. Firstly, we determine the stations, that need to share resources between each other. For clarity, we propose to construct a graph where vertices represent base stations and links between them – a need of mutual resource sharing. We shall call it *an interference graph.*

Usually signal strength or interference level at user device are used as a criterion of resource division need. We intentionally use the definition based on the calculation of the path loss difference (see eq. (3)). It is caused by the specifics of scenarios with non-uniform deployment of small base stations. The point is that having a macrocell close to small cells and employing load balancing mechanisms could lead to a situation where the level of interference or signal strength at the user device may not carry any important information for resources sharing within a network of small cells. In these cases, the assumption that every user is always associated with the station with strongest signal strength is often unacceptable [1].

Let us further assume that it is necessary to share resources between two stations A and B if at least one of them has a user for whom the following condition is satisfied:

$$deltaPl = 10 \log \frac{Pl_A}{Pl_B} < Threshold\,, \tag{3}$$

where Pl_A is the path loss to the station A, and $Threshold$ [dB] is the parameter. Note that the increase in the $Threshold$ value increases the connectivity of the resulting interference graph.

The Figure 2 shows the distribution of $deltaPl$ value for the rightmost station. Point on the plane corresponds to the minimum $deltaPl$ value for its users.

Note, that in case of algorithm implementation within 3GPP LTE standard it is possible to use RSRP measurement reports [11] to get Pl_A and Pl_B values.

Obviously, the topology of the interference graph essentially depends on the $Threshold$. Smaller values correspond to sparse or completely disconnected graph, higher values correspond to denser graph (see Figure 3).

Step II: Resource allocation. After construction of the interference graph we must distribute resources between the stations connected with an edge so that they do not use the same resource blocks. Our algorithm is based on the idea of graph coloring. Each available resource block is associated with a color (terms "resource block" and "color" are further used interchangeably). Graph coloring is performed in a distributed manner by all the base stations which have neighbors in the interference graph. Simply speaking, each station continuously chooses random colors and takes precautions to not use the colors chosen by its neighbors. The proposed coloring algorithm is described in Figure 4.

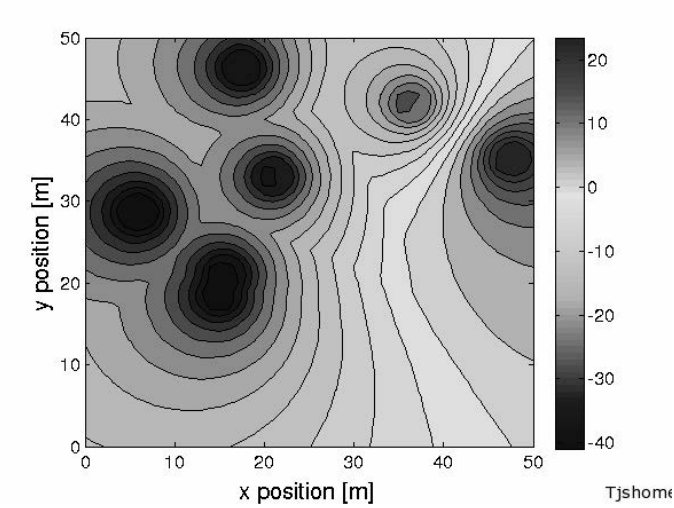


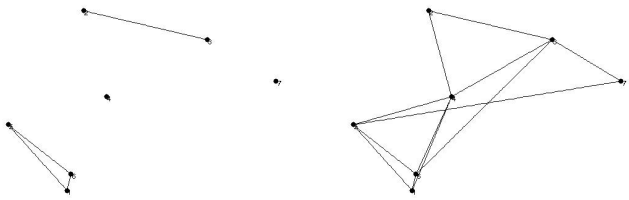
Fig. 2. The distribution of *deltaPl* [dB] value over the users of the rightmost station

One pass through the specified sequence of actions is called coloring round. The final set of colors captured by stations is obtained after several rounds. For the large N_{min} values the algorithm may not converge for any number of the coloring rounds. In this case, the limit on the maximum number of rounds N_{max_rounds} can help reaching a sub-optimal coloring pattern in a reasonable amount of iterations.

Ideally, during the resource allocation process we should take into account 1) chromatic number of a graph, 2) fairness of resource distribution, 3) stations load, and 4) efficiency of the resource usage. It is assumed that in order to achieve an efficient operation mode it is enough to correctly choose the parameters of the coloring algorithm: N_{min} and p_{greedy} (see Figure 4). For example, in practice it appeared effective to reduce parameter p_{greedy} inversely proportional to the size of *captured_colors* list.

In case of algorithm implementation within 3GPP LTE standard it is possible to use the RNTP field of the LOAD INFORMATION message (see [12]) to inform neighbors about resource capturing. For third-party stations this message is interpreted as a signal of potential interference increase in this resource block. This fact allows us to claim that the proposed scheme is compatible with 3GPP LTE standard.

Step III: Resource usage. At this stage all the stations in the network have marked the part of the available resource blocks as captured. The next step is to set the proper constraints on the usage of these resource blocks. The fact that after the stage of resource distribution, station A (with adjacent station B) has captured a particular part of the spectrum indicates the following. Station B should not schedule its users or reduce maximum allowable transmit power on the resource blocks captured by station A.



(a) Threshold = 4 dB (b) Threshold = 10 dB

Fig. 3. Variation of graph topology with *Threshold* parameter

N_{min} – minimum number of colors each base station tends to have
p_{greedy} – rate of greedy color capturing

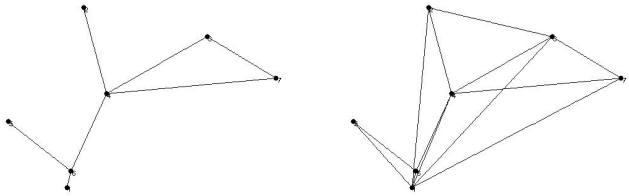
1: **procedure** COLORING_ROUND
2: **if** size (*captured_colors*) $< N_{min}$ **then**
3: *captured_colors* ← random color
4: **else**
5: with probability $(1 - p_{greedy})$ go to step 9
6: $c \leftarrow$ random color not captured by neighbors
7: *captured_colors* ← c
8: **end if**
9: broadcast *captured_colors* to all neighbors
10: exclude neighbors' colors from *captured_colors*
11: **end procedure**

Fig. 4. Coloring Algorithm

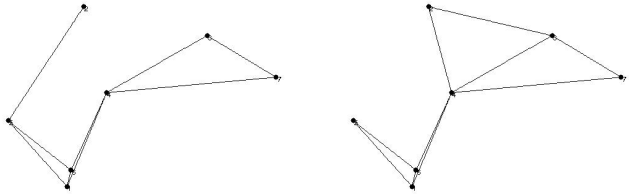
3.2 Enhanced Scheme - Per-TTI Decision

We emphasize once again that all the control messages for graph coloring algorithm are not transmitted by the radio channel but by the backhaul X2 interface. The backhaul links between small base stations can be arranged in different ways (e.g. wired or wireless). High-quality backhaul links has the potential to improve the performance of the algorithm described previously.

We suggest to consider re-coloring of graph each time after the scheduling stage (TD-PS, Time Domain Packet Scheduling stage). During TD-PS stage MAC scheduler of base station chooses the subset of all users to be serviced in the next TTI. Thus, while the construction of an interference graph we propose to consider *only those users* whose data will be sent in the next TTI. Figure 5 demonstrates variation of graph topology with time domain scheduling decisions.



(a) TTI = 1 (b) TTI = 2



(c) TTI = 3 (d) TTI = 4

Fig. 5. Variation of graph topology with time domain scheduling decisions

The rough estimate of the backhaul throughput can be calculated as follows:

$$Rate_{req} = \frac{N_{RB} \times N_{Nbr} \times N_{max\_rounds}}{T} \text{ (bit/s)} \tag{4}$$

where N_{RB} is the number of resource blocks, N_{Nbr} is the average number of neighbors, N_{max_rounds} is the maximum number of coloring rounds and T is the coloring period (1 ms in this case).

In the operating mode of the algorithm this value can rise to tens of Mbit per second with a sub-millisecond latency required. It is obvious that the majority of backhaul types cannot provide such throughput levels. The proposed extension of the algorithm should be considered as a basis for the creation of less overhead-intensive version.

4 Simulation Results

In this section, we study the performance of the proposed schemes by computer simulation. The simulation setup follows closely the suggestions given for the 3GPP LTE evaluation. The simulated small cell scenario is indoor enterprise model. Figure 6 shows as an example the topology of small stations and users (5x5 Grid, see [10]). Detailed simulation parameters are listed in Table 1. As a simulation tool we use a modified version of Vienna LTE Simulator – MATLAB-based system level simulator [13].

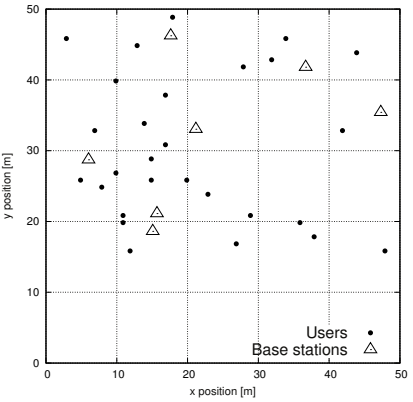


Fig. 6. Small cell topology example

Table 1. Simulation Setup

Scenario Parameters	
Number of Cells, N	7
Number of users, m	28
Freq. Reuse Factor	from 1 to N
Topology	5x5 Grid (see [10])
Traffic type	Full buffer
Scheduler	Proportional Fair
Number of RBs, N_{RB}	100
Channel BW, MHz	20
Path Loss (dB)	$Pl = 37 + 20 \log_{10} d$
Power Control	
Center TX power, P_0	23 dBm
Edge TX power, P_1	0 mW
TN Density, N_0	-174 dBm/Hz
Coloring algorithm	
N_{min}	15
p_{greedy}	$1/\lvert\text{size}(captured\_colors) - N_{min}\rvert$
N_{max_rounds}	100

In the first example, we are going to investigate the impact of graph connectivity in the spectral efficiency. We compare the proposed algorithm, its enhanced per TTI scheme and two well-known reference schemes – 1-Reuse and N-Reuse. Figure 7 shows the dependence of the overall spectral efficiency on the

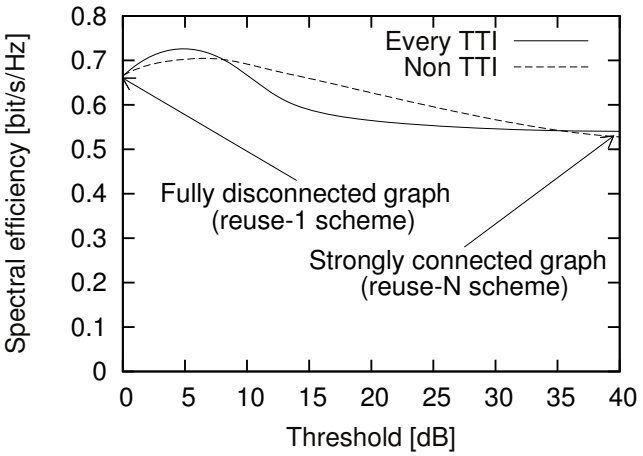


Fig. 7. Overall spectral efficiency

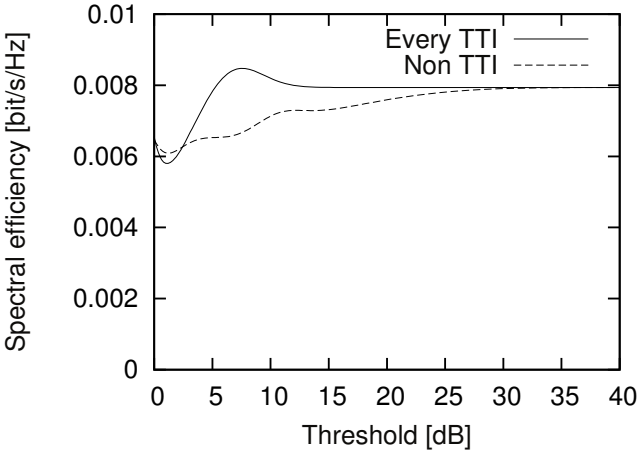


Fig. 8. 5^{th} quantile spectral efficiency

Threshold parameter. Figure 8 shows the same relation for the 5^{th} quantile. Recall that the connectivity of the interference graph strongly depends on the *Threshold* parameter. Value $Threshold = 0$ dB corresponds to a fully disconnected graph (Reuse-1 scheme) and value $Threshold > 40$ dB corresponds to a strongly connected graph (Reuse-N scheme).

It is evident that for both the proposed algorithm and for its time-domain extension, there is an optimal value of *Threshold* parameter (about 5-6 dB). For optimal configuration of parameters we observe capacity growth for users with poor channel conditions (within 30–50%, depending on a reference method). Moreover it is accompanied by increase of the overall system capacity (within 10-30%).

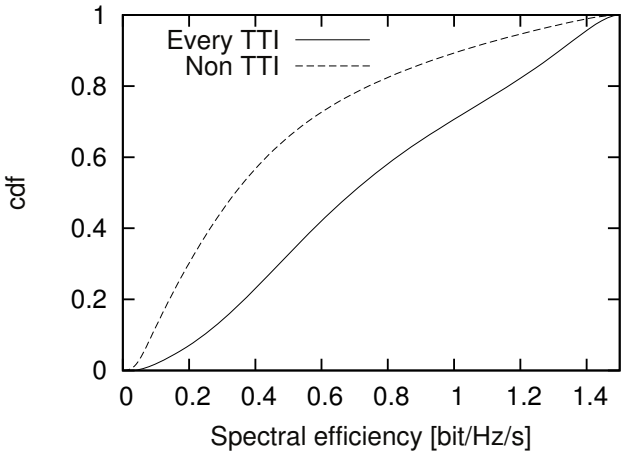


Fig. 9. CDF for system spectral efficiency

Table 2. Peak Performance

	Overall capacity, bit/s/Hz	Cell edge throughput, $10^{-3}\times$ bit/s/Hz
1-Reuse	0.66	6.5
Coloring	0.7	8.7
Per-TTI Coloring	0.73	8
N-Reuse	0.54	8

The optimal *Threshold* value may vary depending on the topology. The description of adaptive control of *Threshold* value is out of scope of this paper. Here we confine ourselves to the statement that *Threshold* value around 5–6 dB is a typical optimum for small cell topologies proposed by 3GPP. Note also that one of the advantages of the proposed scheme is the ability of flexible transition between different interference control policies. The next example compares system spectral efficiency between the proposed algorithm and its time-domain extension. The cumulative distribution function of the overall system efficiency is shown in Figure 9. It should be noted that for Per-TTI algorithm we observe a noticeable improvement for users experiencing poor radio conditions.

Peak performance values for the proposed and standard schemes are summarized in Table 2.

5 Conclusion

We propose to use path loss difference as a measure of interference influence and demonstrate its effectiveness in resource allocation process. We provide the

distributed adaptive algorithm for interference control tailored for the use within small cell LTE environment. We also propose a dynamic extension for the algorithm, which allows scaling it effectively in a time domain. It allows to achieve greater throughput performance for users experiencing poor radio conditions and to control fairness of resource distribution flexibly. We demonstrate the presence of an optimum parameter values for which the throughput performance of the users experiencing poor channel conditions is growing without compromising overall network throughput.

In our work, we demonstrate the effectiveness of path loss difference as a property that characterizes the need to divide resources between small base stations in a non-uniform deployment scenario.

One of the advantages of the proposed algorithms is the possibility of flexible transition between different policies of interference control. The proposed algorithms can behave similar to the convenient frequency reuse schemes (1-Reuse and N-Reuse) by the proper choice of the parameters. The selection of optimal parameters allows to achieve the performance increase for users experiencing poor channel conditions (up to 30-50%, depending on a reference method) which is accompanied by the increase of overall throughput (around 10-30%).

Distributed optimization of the algorithm's parameters is a promising direction of future development. We also want to emphasize another interesting direction – the development of the time-domain extension of the proposed algorithm tailored for realistic backhaul.

Acknowledgment. The authors would like to thank Pavel Boyko, Denis Fakhriev and Mikhail Yakimov of Telum for their comments and fruitful discussions.

References

1. Ghosh, A., Mangalvedhe, N., Ratasuk, R., Mondal, B., Cudak, M., Visotsky, E., Thomas, T., Andrews, J., Xia, P., Jo, H., Dhillon, H., Novlan, T.: Heterogeneous cellular networks: From theory to practice. IEEE Communications Magazine 50(6), 54–64 (2012)
2. T.S.C. Forum, Small cell market status - informa (2012), `http://www.smallcellforum.org/resources-white-papers`
3. Necker, M.C.: Coordinated fractional frequency reuse. In: Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems, MSWiM 2007, pp. 296–305. ACM, New York (2007)
4. Chang, Y.-J., Tao, Z., Zhang, J., Kuo, C.-C.: A graph-based approach to multi-cell ofdma downlink resource allocation. In: IEEE Global Telecommunications Conference, IEEE GLOBECOM 2008, November 30-December 4, pp. 1–6 (2008)
5. Chang, R., Tao, Z., Zhang, J., Kuo, C.-C.: A graph approach to dynamic fractional frequency reuse (FFR) in multi-cell OFDMA networks. In: IEEE International Conference on Communications, ICC 2009, pp. 1–6 (June 2009)
6. Alam, S.M.N., Haas, Z.J.: Coverage and connectivity in three-dimensional networks. In: Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, MobiCom 2006, pp. 346–357. ACM, New York (2006)

7. Mustafa, H.E.-B., El-Tantawy, M.M., Aboul Dahab, M.: Performance evaluation of frequency reuse schemes in LTE based network
8. Chiu, C.-S., Huang, C.-C.: Combined partial reuse and soft handover in OFDMA downlink transmission. In: IEEE Vehicular Technology Conference, VTC Spring 2008, pp. 1707–1711 (May 2008)
9. Li, G., Liu, H.: Downlink radio resource allocation for multi-cell ofdma system. IEEE Transactions on Wireless Communications 5(12), 3451–3459 (2006)
10. Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancements for E-UTRA physical layer aspects (Release 9), 3GPP TR.36.814
11. Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation, 3GPP TS.36.211
12. Evolved Universal Terrestrial Radio Access Network (E-UTRAN); X2 Application Protocol (X2AP), 3GPP TS.36.423
13. Mehlfuhrer, C., Colom Ikuno, J., Simko, M., Schwarz, S., Wrulich, M., Rupp, M.: The Vienna LTE simulators - enabling reproducibility in wireless communications research. EURASIP Journal on Advances in Signal Processing 2011(1), 29 (2011), http://asp.eurasipjournals.com/content/2011/1/29

Smart Cities Software: Customized Messages for Mobile Subscribers

Manfred Sneps-Sneppe[1] and Dmitry Namiot[2]

[1] Ventspils University College
Ventspils International Radioastronomy Centre
Ventspils, Latvia
manfreds.sneps@gmail.com
[2] Lomonosov Moscow State University
Faculty of Computational Mathematics and Cybernetics
Moscow, Russia
dnamiot@gmail.com

Abstract. This paper introduces a new way of delivering local messages to mobile subscribers. Our application presents a mashup from passive monitoring for smart phones and cloud-based messaging for mobile operational systems. Passive monitoring can detect the presence of mobile phones without active participation from the users. It does not require prior calibration, nor does it require mobile users to mark their own location on social networks (like traditional check-ins). Mobile users do not need to run location track applications on their phones the. At the same time, a production-based expert system built around cloud messaging allows interested parties to directly deliver their custom information to mobile users in proximity.

Keywords: location, proximity, Wi-Fi, Bluetooth, cloud, messages.

1 Introduction

The classical definition for the term 'context-aware' [1] describes context as a location, identities of nearby people and objects, and changes to those objects. Most of the modern authors define context awareness as a complementary element to location awareness. Location serves as a determinant for the main processes, and context adds more flexibility with mobile computing and smart communicators [2].

There are many practical use cases, where the concept of location can be replaced by that of proximity. On one hand, this applies to use cases where the detection for exact location is difficult, even impossible, or not economically viable [2]. Very often, this replacement is related to privacy. For example, a privacy-aware proximity-detection service determines if two mobile users are close to each other, without requiring them to disclose their exact locations [3]. Proximity can be used as a main formation for context-aware browsers [4]. In this concept, any network node (e.g., Wi-Fi hot spot or Bluetooth node), could be used as a presence trigger. This trigger can open access to some content, discover existing content, as well as cluster nearby

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 25–36, 2013.

mobile users. As per other developed algorithms for privacy-aware proximity-detection methods, we can mention papers [5–7], for example. Technically, the final goal for the proposed systems is to allow two online users to determine if they are close to each other, without requiring them to disclose their exact locations to a service provider, or other friends. There are different approaches to privacy protection. We can mention, for example, the anonymization of location tracks, degrading the accuracy of location data, and obfuscation (through which the actual coordinates are substituted by street or city level information). We can add the k-anonymity concept [8], as a set of spatial and temporal cloaking algorithms. As per this concept, the location can be cloaked among those of k-1 others by choosing either a large enough area to contain at least k users, or a temporal window, such that a selected area is crossed by at least k users. Our own approach [8] proposes a new paradigm: keep location data anonymous, and share identification data on a peer to peer basis. The WATN (Where Are They Now) model creates a special form of distributed database that splits location info and identity information. In this distributed data store, identity info is always saved locally where the social graph data store is still centralized.

We should note also that, privacy protection can lower the quality of service or even make it completely irrelevant. For example, mobile services which include navigation become useless in cases of location obfuscation, etc. It means, privacy problems should be tight-linked with main functionality for mobile services. Context awareness has been utilized to improve the awareness of users about their privacy risks and settings in ubiquitous computing systems [9]. Privacy preferences are typically formalized in context-aware privacy policies.

Very often the main function for context-aware systems is to generate proximity messages when friends approach each other closer than some predefined distance threshold. Technically, this threshold can be defined individually for each user (or group of users). There are several important remarks.

On one hand, the word “friends” should be used in the extended meaning. They could be either friends, as per some social network (social circle), or simply other participants (keeping in mind the possible privacy-based anonymization, for instance).

At the second, the term “distance” here depends on the metric used for the measurements. The classical anti-pattern example includes the shortest metric path and two users on different sides (banks) of the river. The distance between users could be within the given threshold, but such “proximity” is useless for pedestrians.

Metric measurements for privacy can be replaced with some approximation by wireless proximity (network proximity). For this paper, network proximity definition is very intuitive. It is a measure of how mobile nodes are close to, or far away from, the elements of a network infrastructure. The classical model uses Bayesian reasoning and a hidden Markov model (HMM) [10]. They took into account not only signal strengths, but also the probability of seeing an access point from a given location. But the biggest practical problem for such models is the mandatory manual calibration phase. Instead of manual calibration, ActiveCampus [11] uses a formula that approximates the distance to an access point as a function of signal strength. Using a

hill-climbing optimization technique, this system computes location to an accuracy of about 10 meters using signal strengths from multiple access points.

Signal strength is based upon knowing the power of the transmitted signal from one node (AP or client) and then measuring the amount of received power at the other node [12]. The common practice is to report the power received in units of decibels relative to milliwatts or dBm. A channel model could be used to convert the power loss between the transmitting and receiving antennas of the two nodes, to estimate the distance between those antennas. The classical channel model used for this application is free-space path loss. This model proposes that the received power drops off as the square of the distance.

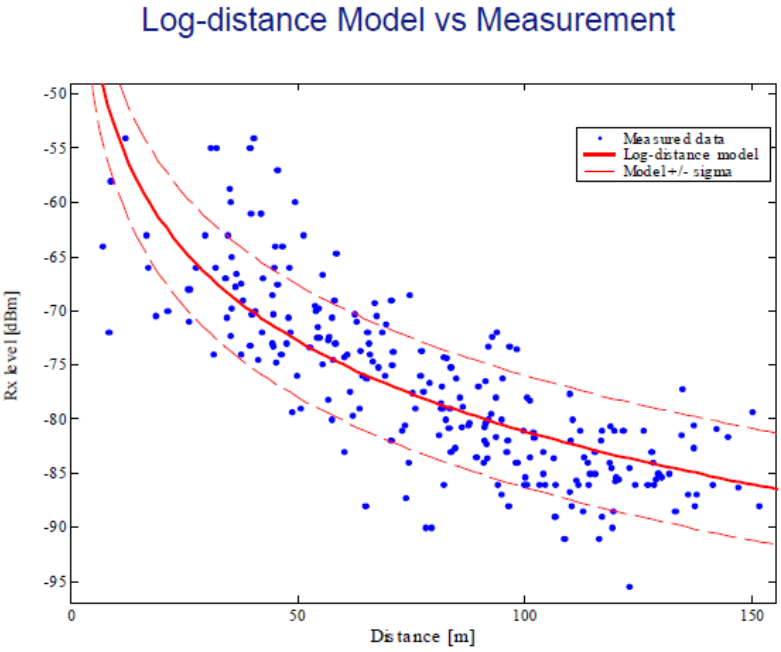


Fig. 1. RSSI measurements vs. model data [13]

Figure 1 shows the real signal strength measurements versus Log-distance model.

There are several systems that can use empirical models for network proximity as a basis for mobile services. On one hand, we can mention here our own system SpotEx (Spot Expert) [14]. According to this model, any existing or even specially-created Wi-Fi hot spot could be used as a presence sensor with associated productions rules. This set of rules can trigger access for some user-generated information snippets. SpotEx presents an especially developed context-aware browser, which can present that information to mobile subscribers.

The rest of the paper is organized as follows: Section II describes a passive monitoring for mobile devices. In Section III we describe Cloud Messaging platforms. In Section IV we discuss our mashup for passive monitoring and Cloud Messaging.

2 Wi-Fi Devices and Monitoring

The typical, Wi-Fi based, location-monitoring application uses collected database of so called Wi-Fi “fingerprints”, including MAC addresses and the received signal strengths (RSSI) from nearby access points. This database could be used for Wi-Fi based positioning, as well as for discovering the user's behavioural patterns [15]. A classical approach to Wi-Fi fingerprinting [16] involves RSSI (signal strength). The basic principles are transparent. At a given point, a mobile application may hear (“see”) different access points with certain signal strengths. This set of access points and their associated signal strengths represents a label (“fingerprint”) that is unique to that position. The metric that could be used for comparing various fingerprints is k-nearest-neighbours in signal space. It means that two compared fingerprints should have the same set of visible access points. As the next step, they could be compared by calculating the Euclidian distance for signal strengths. At the same time, the need for the collection of fingerprints is the biggest problem for this approach. Problems associated with the collection of fingerprints are fairly obvious. It is the price of the calibration process, the need for rework after the changes in the network and, most importantly, a lack of support for dynamic networks.

In our new service we’ve decided to use one of the fingerprints-less models. It involves sniffing for beacon frames. Typically, during the calibration phase of Wi-Fi fingerprinting, beacon frames are collected from nearby access points at each survey position. As the next step, the MAC address, RSSI (signal strength), and timestamp could be extracted from each beacon. In our system, we use the reverse schema. We prefer to analyze beacons transmitted by Wi-Fi devices (Wi-Fi clients).

Collecting traces of Wi-Fi beacons is a well-known approach for getting the locations of Wi-Fi access points. Beacon frames are used to announce the presence of a Wi-Fi network. As a result, an 802.11 client receives the beacons sent from all nearby access points. The client receives beacons even when it is not connected to any network. In fact, even when a client is connected to a specific AP, it periodically scans all the channels to receive beacons from other nearby APs. It lets clients keep track of networks in its vicinity. But at the same time, the Wi-Fi client periodically broadcasts an 802.11 probe request frame. Wi-Fi access point sends back an appropriate probe request response. As per Wi-Fi spec, a station (client) sends a probe request frame when it needs to obtain information from another station. For example, a radio network interface card would send a probe request to determine which access points are within range.

A Probe Request frame contains the SSID, and the rates supported by the mobile terminal. APs that receive Probe Requests use the information to determine whether the mobile terminal can join the network. For joining the network, the mobile station must support all the data rates required by the network. It must want also to join the network identified by the provided SSID. This may be set to the SSID of a specific network, or set to join any compatible Wi-Fi network. Software drivers that allow cards to join any network use the broadcast SSID in Probe Requests [17].

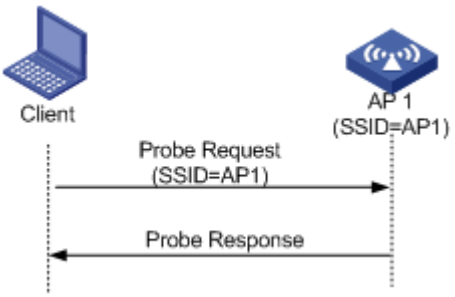


Fig. 2. Probe request/response

Technically, probe request frame contains the following information:

- source address (MAC-address)
- SSID
- supported rates
- additional request information
- extended support rates
- vendor specific information

Our access point can analyze received probe requests. Obviously, any new request (any new MAC-address) corresponds to a new wireless customer nearby.

For sending probe requests, the network interface on the client side works in active mode. But for the client itself, it could be treated as a passive mode. The client does not require load specific applications, or run specific applications. In other words, there is no client-side activity. These systems use only common, off-the-shelf access point hardware to both collect and deliver detections. Thus, in addition to high detection rates, it potentially offers very low equipment and installation cost [18]. Note that Bluetooth devices could be monitored by the same principles.

Wi-Fi based device detections are made by capturing Wi-Fi transmissions from the device in question. This detection uses only a part from the above-mentioned probe request. It is a device-unique address (MAC address). This unique information lets us re-identify devices (mobile phones) across our monitors.

We should also note that, passive Wi-Fi detection is not 100% reliable. In general, mobile phones (mobile OS, actually) can transmit probe requests at their discretion. As per detection rate, we can cite [18] and our own estimations. On average, if Wi-Fi is turned on, a monitor detects a passing smartphone 70% of the time (75%-80% in our own experiments).

There are commercial off-the-shelf components that can provide passive Wi-Fi monitoring. For example, Navizon I.T.S. [19]. These devices can be detected without the need of being connected to a specific access point, enabling the detection of any smartphone, laptop or hands-free device which is in the coverage area.

The information read from each user contains:

- the MAC address of the wireless interface, which allows to identify it uniquely
- the strength of the signal (RSSI), which gives us the average distance of the device from the scanning point

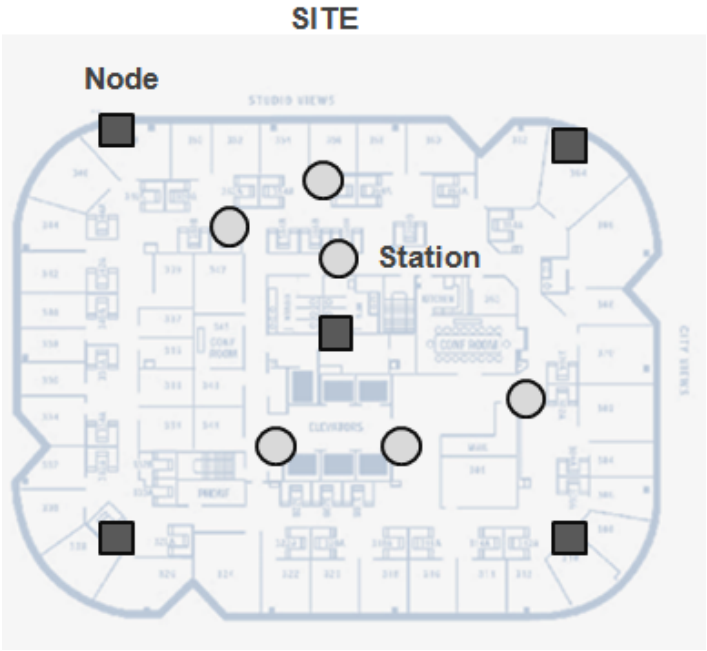


Fig. 3. Smartphones detection

Note: The key moment here is MAC-address for mobile device. We will use it for re-identification only. It means, that for maintaining privacy, we do not need to store an original address in our database. It is enough just to keep some hash-code for this address.

The typical tasks this approach could be applied to are:

- get a number of people passing daily in the streets
- detect an average time of the stance of the people in a street or in a building
- differentiate between residents (daily matches) and visitants (sporadic matches)
- detect the walking routes of people in shopping malls, and the average time in each area

In general, it could be described as a real analytics for real places. It is what makes Google Analytics for web sites, but applied to real places and real visitors.

In this paper we propose a new model (use case) for passive monitoring. It is messaging for real places and real visitors.

3 Cloud Messaging

Cloud Messaging for mobile platforms is a service that allows data providers to send data from own servers to mobile devices.

For example, Google Cloud Messaging (GCM) service handles all aspects of the queuing of messages and delivery to the target Android application running on the target device [20]. Apple Push Notification Service (APN) is a robust and highly efficient service for propagating information to devices such as iPhones, iPads, and iPod touch devices. As per APN model, each device should establish an encrypted IP connection with the service. This persistent connection will be used for devivering notifications. It is important that the mobile user does not need to always keep the APN-enabled application in the running state. The device can alert the user that the application has data waiting for it. So, the notifications for an application can arrive when that application is not running [21].

We can also mention here: Microsoft's Push Notification Service (MPNS) for Windows Mobile [22], Blackberry's Push Service (BPS) [23], and Nokia's Notifications API (NNA) for Symbian and Meego devices [24].

Architectures of these push-notification services have common features. On one hand, application servers send a notification message with an intended receiver (or the target mobile device) to one of the cloud-based messaging servers. Messaging servers push the message to the target mobile device. The push-notification service eliminates the need for application servers to keep track of the state of a mobile device (i.e., online or offline). Furthermore, mobile devices do not need to periodically probe (poll) the application servers for messages. It reduces the workloads of the application servers, and greatly simplifies the mobile application development.

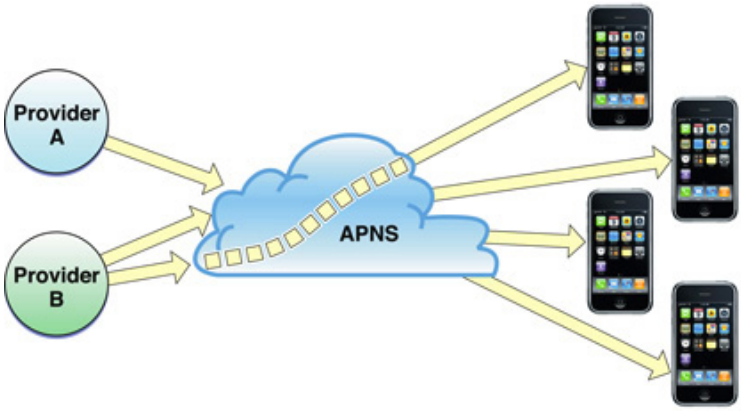


Fig. 4. APN architecture

As per the APN concept, this service transports and routes a notification. APN delivers the notification from a given provider to a given device. Actually, it is true for all cloud messaging services. A notification is a short message which consist of the device token and so-called payload. The device token is some unique ID analogous to a phone number or MAC-address. This unique ID enables APN to locate the device on which the client application is installed. APNs can also use it to authenticate the routing of a notification. The payload is some data array associated

with the notification. Technically, it is a JSON-defined property list. It specifies how the user of an application is to be alerted.

Any APN-enabled application must be registered to receive push notifications. For example, it could be performed right after APN-enabled application is installed on a device. Operational system (iOS in case of Apple) receives the registration request from an application and connects with APNs. APNs generate a device token. This process uses information contained in the unique device certificate. Firstly, APN encrypts the device token with a token key. Then encrypted key should be returned to the device. This information flow is shown in Figure 5.

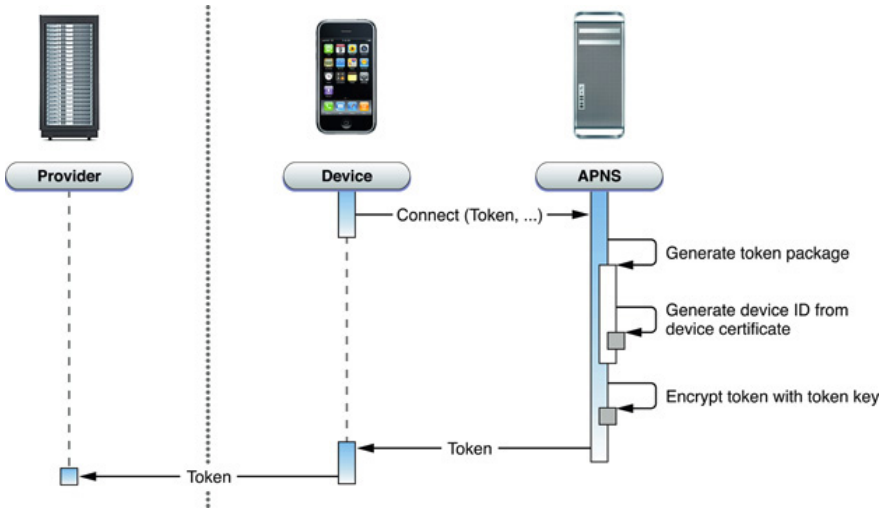


Fig. 5. Subscription to APN service

Mobile device must provide APNs with the token every time it connects. APNs decrypts the device token and validates it. APNs check that the token was generated for the connecting device. APN can check and compare the device identifier contained in the token and the device identifier in the device certificate.

Every notification that any provider sends to APNs for delivery to a device must contain the device token it obtained from an application on that device. So, any provider must keep the list of tokens (actually, the list of subscribers). APN can decrypt the token using the token key. It proves that the notification is valid. For valid notification APN uses the device token to determine the destination device for this message [21].

The most interesting fact for future development is the need for the registration. A registration request should originate from client side application. It means that, this application can read the system settings for the mobile device. Actually, the only data we are interested in for the future development is the MAC-address.

4 Messaging Service

Based on the above-mentioned description, we can make a simple assumption. The registration process for cloud messaging enabled application should include a MAC-address. This decision lets us simply compare subscription info with the locally-detected (presented) mobile subscribers in passive monitoring. It is the main idea behind our approach - how to combine (mashup) two existing services: Cloud Messaging and passive monitoring.

The whole schema is actually very transparent.

1) The mobile user informs Cloud Messaging about his intention to receive messages.

2) Messages are divided by topics. Each topic actually corresponds to some location area with passive Wi-Fi monitoring.

3) Our sender (provider) saves the registration ID, topic and MAC-address in a central database. MAC-address will link together virtual subscribers and physical visitors.

4) Wi-Fi monitoring detects the presence of mobile phones.

5) Our daemon scans detection log, extracts MAC-addresses and compares them with the subscription database

6) As soon as the subscriber is detected (he is somewhere nearby) we can use CGM for delivering some custom messages

It is illustrated on Figure 6.

In step 3, the application (subscriber) can provide a token and a MAC-address. Note that the MAC-address in this schema is used for re-identification only. So, to maintain privacy, we can replace it with some hash-code (and it is true for both processes: monitoring and subscription).

The typical use cases for this approach are proximity marketing and news delivery in Smart City applications. It is some like geo-fence concept, but redefined in the network proximity terms.

As per the messaging server itself, we follow the same idea as in the SpotEx application [25]. The messaging server could be presented as a server-side implementation of SpotEx.

What are the advantages of this approach? On one hand, it is based on passive monitoring. We do not need to develop some special applications for mobile subscribers, nor do we need to ask for any special actions from them, like running some application, checking-in in social networks, etc. There is no need to promote (distribute) client-side applications across mobile subscribers. The messaging will automatically target subscribers physically present in the covered area. The process for subscription and un-subscription is very straightforward. The “check-in” process (passive discovering) is secure. It does not keep records in social networks like ordinary check-ins in Foursquare, Facebook, etc. It is secure and does not require user’s identification.

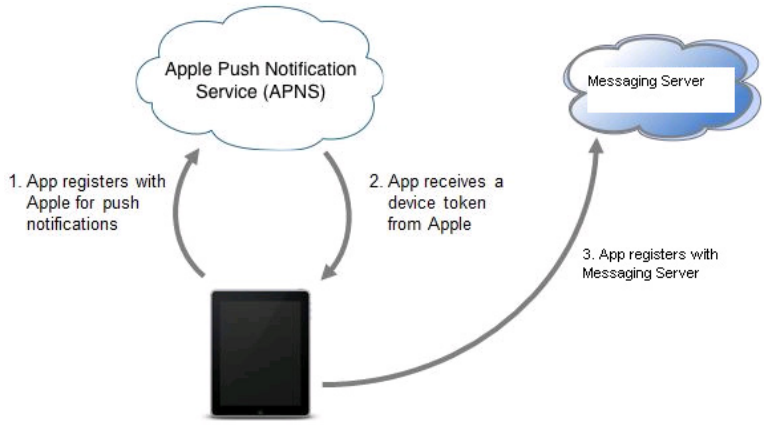


Fig. 6. Messaging architecture

What are the disadvantages? On one hand, the passive monitoring (as we wrote above) is not 100% reliable. Push-messaging delivery requires internet connectivity. But, at the same time, installing active Wi-Fi access point on-site (so mobile users can connect), will improve the discovery process.

We evaluate the first practical implementation of the described approach in Lomonosov Moscow State University.

5 Conclusion

In this article, we present a solution that combines passive Wi-Fi monitoring for mobile devices and cloud-based notifications. Passive monitoring uses probe requests from Wi-Fi specifications for detecting nearby clients. Notification module uses cloud messaging (push notifications) from mobile operational systems. This approach does not require special client-side applications, nor does it require a publishing location information in the social networks. The proposed approach automatically guaranties that custom messages will target online subscribers in the nearby area only. Practical use cases for this application are proximity marketing and Smart City projects.

Acknowledgement. The paper is financed from EDRF's project SATTEH (No. 2010/0189/2DP/2.1.1.2.0./10/APIA/VIAA/019) being implemented in Engineering Research Institute “Ventspils International Radio Astronomy Centre” of Ventspils University College (VIRAC).

References

[1] Schilit, G., Theimer, B.: Disseminating Active Map Information to Mobile Hosts. IEEE Network 8(5), 22–32 (1994)

[2] Namiot, D.: Context-Aware Browsing – A Practical Approach. In: 2012 6th International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST), pp. 18–23 (2012), doi:10.1109/NGMAST.2012.13

[3] Šikšnys, L., Thomsen, J., Šaltenis, S., Yiu, M.: Private and Flexible Proximity Detection. In: 2010 Eleventh International Conference on Mobile Social Networks, Mobile Data Management (MDM), pp. 75–84 (2010)

[4] Namiot, D., Sneps-Sneppe, M.: Using Network Proximity for Context-aware Browsing. International Journal on Advances in Telecommunications 5(3 and 4), 163–172 (2012)

[5] Scipioni, M.P., Langheinrich, M.: To Share or Not To Share? An Activity-centered Approach for Designing Usable Location Sharing Tools. In: Workshop on Usable Privacy & Security for Mobile Devices (U-PriSM) (2012), Co-located with Soups 2012

[6] Ruppel, P., Treu, G., Küpper, A., Linnhoff-Popien, C.: Anonymous User Tracking for Location-Based Community Services. In: Hazas, M., Krumm, J., Strang, T. (eds.) LoCA 2006. LNCS, vol. 3987, pp. 116–133. Springer, Heidelberg (2006)

[7] Toch, E., Cranshaw, J., Drielsma, P., Tsai, J., Kelley, P., Springfield, J., Cranor, L., Hong, J., Sadeh, N.: Empirical models of privacy in location sharing. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, pp. 129–138. ACM (2010)

[8] Namiot, D., Sneps-Sneppe, M.: Where Are They Now – Safe Location Sharing. In: Andreev, S., Balandin, S., Koucheryavy, Y. (eds.) NEW2AN/ruSMART 2012. LNCS, vol. 7469, pp. 63–74. Springer, Heidelberg (2012)

[9] Schaub, F., Damiani, M., Malin, B.: Mobility Data Mining and Privacy. Seminar, August 12-17 (2012), http://www.dagstuhl.de/12331, doi: 10.4230/DagRep.2.8.16

[10] Ladd, A.M., et al.: Robotics-Based Location Sensing using Wireless Ethernet. In: Eighth International Conference on Mobile Computing and Networking, Atlanta, GA, USA (2002)

[11] Griswold, W.G., et al.: ActiveCampus - Sustaining Educational Communities through Mobile Technology, p. 19. University of California, San Diego (2002)

[12] Golden, S., Bateman, S.: Sensor Measurements for Wi-Fi Location with Emphasis on Time-of-Arrival Ranging. IEEE Transactions on Mobile Computing 6(10), 1185–1198 (2007)

[13] Laitinen, H.: WLAN Location Methods. Graduate Course Slides. Univ. of Finland (April 2004), http://www.comlab.hut.fi/opetus/333/2004slides/topic33.pdf

[14] Namiot, D., Schneps-Schneppe, M.: About location-aware mobile messages. In: International Conference and Exhibition on Next Generation Mobile Applications, Services and Technologies (NGMAST), pp. 48–53 (2011), doi:10.1109/NGMAST.2011.19

[15] Rekimoto, J., Miyaki, T., Ishizawa, T.: LifeTag: WiFi-based continuous location logging for life pattern analysis. In: Hightower, J., Schiele, B., Strang, T. (eds.) LoCA 2007. LNCS, vol. 4718, pp. 35–49. Springer, Heidelberg (2007)

[16] Chen, Y., Chawathe, Y., LaMarca, A., Krumm, J.: Accuracy characterization for metropolitan-scale Wi-Fi localization. In: ACM MobiSys (2005)

[17] Gast, M.: 802.11 Wireless Networks: The Definitive Guide, 654 p. O'Reilly Media, Inc. (2005)

[18] Musa, A., Eriksson, J.: Tracking Unmodified Smartphones Using Wi-Fi Monitors. In: SenSys 2012, Toronto, November 6-9 (2012)

[19] Navizon I.T.S., http://its.navizon.com/doc/index.html (retrieved: January 2013)

[20] Google Cloud Messaging, http://developer.android.com/google/gcm/index.html (retrieved: January 2013)
[21] Apple Push Notification Service, http://developer.apple.com/library/mac/#documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/ApplePushService/ApplePushService.html (retrievd: January 2012)
[22] Microsoft Inc. Push Notifications Overview for Windows Phone, http://msdn.microsoft.com/en-us/library/windowsphone/develop/ff402558(v=vs.105).aspx (retrived: January 2013)
[23] Research In Motion Inc. Blackberry push service, http://us.blackberry.com/developers/platform/pushapi.jsp (retrived: January 2013)
[24] Nokia Inc. Notifications API, https://projects.developer.nokia.com/notificationsapi/wiki (retrived: January 2013)
[25] Namiot, D., Sneps-Sneppe, M.: Context-aware data discovery. In: IEEE 2012 16th International Conference on Intelligence in Next Generation Networks (ICIN), pp. 134–141 (2012), doi:10.1109/ICIN.2012.6376016

802.11 Buffers: When Bigger Is Not Better?*

David Malone[1], Hanghang Qi[1], Dmitri Botvich[2], and Paul Patras[1]

[1] Hamilton Institute, National University of Ireland Maynooth
[2] TSSG, Waterford Institute of Technology, Ireland

Abstract. While there have been considerable advances in the modelling of 802.11's MAC layer in recent years, 802.11 with finite buffer space is considered difficult to analyse. In this paper, we study the impact of finite buffers' effect on the 802.11 performance, in view of the requirements of interactive applications sensitive to delay and packet loss. Using both state-of-the art and simplified queueing models, we identify a surprising result. Specifically, we find that increased buffering throughout an 802.11 network will not only incur delay, but may actually increase the packet loss experienced by stations. By means of numerical analysis and simulations we show that this non-monotonic behaviour arises because of the contention-based nature of the medium access protocol, whose performance is closely related to the traffic load and the buffer size. Finally, we discuss on protocol and buffer tuning towards eliminating such undesirable effect.

1 Introduction

The IEEE 802.11 protocol has grown to be the de facto standard for wireless LANs since it was developed and released in the 1990s. While the 802.11 specification includes both centralised and decentralised MAC mechanisms, the distributed coordination function (DCF), a random access scheme based on carrier sense multiple access with collision avoidance (CSMA/CA) through binary exponential backoff (BEB), is the scheme widely used in current devices. Modeling the 802.11 DCF performance and subsequent protocol optimisations have advanced considerably over the last 10–15 years. In particular, Bianchi's technique [3], has permitted the performance analysis of a number of protocol variations. This approach models the MAC protocol states, in particular the backoff counter and backoff stage, as a Markov chain. Precisely, the analysis leads to a Markov chain with $O(1000)$ states and by finding its stationary distribution, one can predict the throughput of the network.

For tractability, Bianchi studied an 802.11 network where stations always had packets to transmit (i.e. saturation conditions). In this situation, queueing dynamics can be ignored, which simplifies the model but also does not capture

* The research leading to these results was partially funded by the European Community's 7th Framework Programme (FP7-ICT-2009-5) under grant no. 257263 (FLAVIA project) and Science Foundation Ireland under grant no. 07/SK/I1216a and 08/SRC/I1403.

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 37–48, 2013.

accurately the protocol behaviour under realistic traffic. Building on this, other authors considered non-saturated networks with small buffers (e.g. [10]) or with infinite buffers (e.g. [12,6]). In the former case, queue state is simple and does not significantly complicate the model. In the latter case, if the analysis of the queue and the MAC are decoupled by assuming they operate probabilistically independently, then both the queueing and MAC parts of the model remain relatively simple, and so the analysis remains tractable.

Performance of 802.11 with finite buffers is commonly considered to be difficult to analyse. If one couples the MAC and the queue, the result is a larger Markov chain whose stationary distribution is hard to calculate explicitly. The usual formulation results in a Markov chain with $O(1000K)$ states, where K is the number of packets that can be buffered. This has resulted in a number of models which are tractable as long as buffering is limited to a modest number of packets. Alternatively, if one decouples the MAC and the queue, the result is a reduction in accuracy of the predictions [7]. Liu et al. [9] introduced a model that retains much of the coupling between the MAC and the queue, while they also provide numerical techniques for efficiently solving for the Markov chain's stationary distribution.

In this paper, we present a study of the 802.11 behaviour with finite-load finite-buffer, by building upon the state-of-the-art Markov chain model of Liu et al. Specifically, we are interested in the case where small buffers may offer a performance advantage to delay-sensitive traffic, such as online games and conversational video/voice. We show that increasing buffer sizes may actually *increase* the packet loss rate, while also increasing delays. To demonstrate that this is not an artefact of the model Liu et al. proposed, we show that the same effect is present also in a much simpler model that combines queueing and contention. Using numerical analysis and simulations, we illustrate that this effect arises due to the contention-based access to the wireless medium. Finally, based on these findings, we discuss on buffer tuning and MAC configuration towards improving the performance of interactive applications.

2 802.11 Modelling with Finite Buffers

To study the impact of the buffer size of the 802.11 performance, we employ two analytical models: the model introduced by Liu et al. [9], which integrates the queueing and MAC operation, and a simplified $M/M/1/K$ station model that we solve.

2.1 Markov Chain Model of Liu et al.

The details of the model proposed by Liu et al. can be found in [9] with an extensive validation. Here we only summarise the key results that we use for our analysis. The Markov chain therein has 3-dimensional state space consisting of 802.11's backoff counter, backoff stage and the current queue length. As in Bianchi's model, the chain is not real-time, but evolves upon counter decrements at the MAC layer. Certain approximations are made for tractability, e.g.

the number of arrivals in a state change is limited to one packet. The authors give a method for solving for the stationary distribution of the Markov chain using linear algebra techniques, which involves considerably lower complexity as compared to brute-force methods.

Of particular interest to our study is the expression of the total delay (due to both queueing and contention), which if we consider frames do not exceed the maximum retry limit, can be computed as [9]:

$$D_t = \Delta \left(1 + \sum_{h=0}^{K} hP[Q = h] \right),$$

where Δ denotes the MAC delay, i.e. the average time elapsed since a packet reaches head-of-the-line until its successful transmission, $P[Q = h]$ defines the probability of the queue being h at any given time, and K denotes the maximum queue length. Further, under the same assumptions the packet loss rate due to buffer being full (i.e. blocking probability) is given by:

$$P_B = P[Q = K].$$

To study the effect of the buffer size on the delay and packet loss performance, we have independently implemented this model. Comparing the output with the figures reported in [9], we confirm the accuracy of our implementation.

2.2 M/M/1/K Simplified Analysis

For comparison, we also consider a simplified model, where we replace the 802.11 DCF MAC with a simple slotted Aloha-style network. We still view each station as a queue of size K, but use an $M/M/1/K$ approximation to model its behaviour. The $M/M/1/K$ approximation assumes that queue inter-arrival times and service times are independent and exponentially distributed, which allows us to explicitly calculate many queue statistics and will prove useful in further validating the findings we obtain by employing the model of Liu et al. Thus we will solve for the probability of a successful transmission in terms of the Aloha transmission probability, the size of the network, the traffic arrival rate and buffer size. Based on this, we derive the blocking probability and total delay, to examine the buffer size impact on the performance.

Suppose we have n queues feeding a slotted Aloha style network where each queue transmits in a slot with probability τ_0 when it has a packet. We will assume the event that a queue transmits in each slot is independent and identically distributed. Let P_{ne} be the probability that a queue is not empty. Let $\tau = P_{ne}\tau_0$ be the probability that each node transmits. The service rate for a queue is the probability that a queue transmits with no collision, or

$$\mu = C(\tau) = \tau_0(1 - \tau)^{n-1},$$

and service times will actually be geometrically distributed.

Now suppose arrivals to the queue occur at a fixed rate of λ packets per slot. Consequently, we have a traffic intensity of $\rho = \lambda/\mu$. Then, approximating the queue by an $M/M/1/K$ model [1], we can compute the queue non-empty probability as

$$P_{ne} = \frac{\rho - \rho^{k+1}}{1 - \rho^{k+1}}.$$

Thus we can regard $P_{ne} = Q(\mu/\lambda)$, remembering that λ is fixed. So, the operating point of the network is given by a fixed point characterised by:

$$\mu = C(\tau) = C(P_{ne}\tau_0) = C(Q(\mu/\lambda)\tau_0).$$

Since τ_0 and λ are given parameters, we can solve this numerically to find μ and then substitute to find P_{ne}.

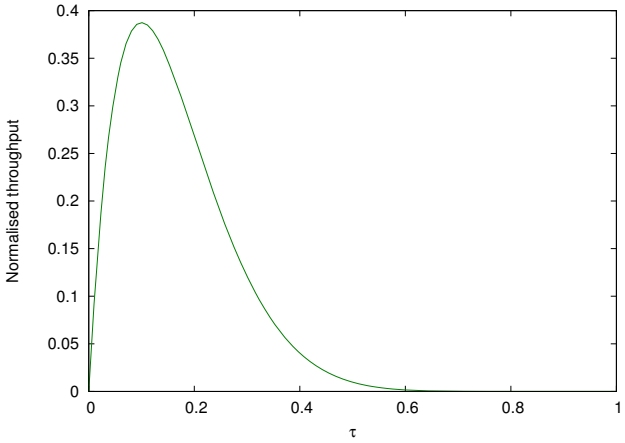


Fig. 1. Throughput for a slotted Aloha network with $n = 10$ stations

Note that for standard slotted Aloha networks, it is well-known [2] that the value of τ which gives optimal throughput is $1/n$. This is illustrated in Figure 1 for $n = 10$ stations, where we can see a clear peak in the total throughput of the network at $\tau = 0.1$. Below this peak, stations do not transmit sufficiently often, leaving too many slots idle, whereas the network experiences an increased number of collisions beyond the peak, which results in reduced throughput.

Having computed μ and knowing λ, we can obtain the packet loss probability due to buffer being full, as follows

$$P_B = \frac{1 - \rho}{1 - \rho^{K+1}}\rho^K.$$

To derive the expression of the total delay a packet undergoes from the time of arrival to the queue until successful transmission, we first compute the average queue length,

$$\bar{N} = \frac{1-\rho}{1-\rho^{K+1}} \sum_{k=0}^{K} k\rho^k$$

and then apply Little's law [8], which yields

$$D_t = \frac{\bar{N}}{\lambda}.$$

3 Performance Analysis

In what follows we analyse the impact of the buffer size on the packet loss and delay performance of the networks, using the models detailed in Sec. 2. Consider a symmetric network, in the sense that all stations are have the same buffer size and operate under identical traffic load. Consequently, we expect each station to experience the same mean throughput, queueing delay and packet loss probability. Unless otherwise stated, we assume a network with $n = 10$ nodes that transmit frames with 500-Byte payload and employ the MAC and PHY layer parameters of IEEE 802.11b as detailed in Table 1.

3.1 Numerical Results with the Model of Liu et al.

First we investigate the packet loss rate and total delay predicted by the 3-D Markov chain model of Liu et al. [9], as the buffer size is varied. Initially we examine the performance under a light load regime, whereby the total offered load is 60% of the network's idealised total capacity, i.e. the number of packets that could be transmitted if the medium was occupied by back-to-back successful transmissions, spaced as closely as permitted by the protocol without backoff, which also the convention used in [9]. Note, however, that the practical capacity is somewhat lower due collisions and backoff procedure.

Figure 2 shows the packet loss rate and delay for this scenario. Since the network is lightly loaded, we observe a familiar pattern, i.e. the delay experienced by each packet is effectively constant. Also, as we increase the available buffer space, the packet loss rate decreases since it is less likely the queue will be full upon the arrival of a new packet.

Table 1. 802.11b MAC parameters

Basic rate	1 Mb/s	$CW_{min} = W$	32
Data rate	11 Mb/s	m ($CW_{max} = 2^m CW_{min}$)	5
Preamble	144 bits	SIFS	10 μs
Headers	40 bytes	DIFS	50 μs
Payload $E[P]$	500 bytes	Slot time σ	20 μs
ACK	14 bytes	Propagation delay	1 μs

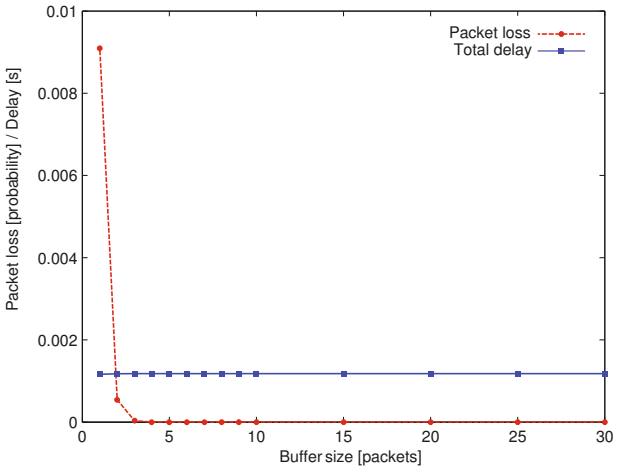


Fig. 2. Delay and packet loss as buffer size is varied. 60% total offered load.

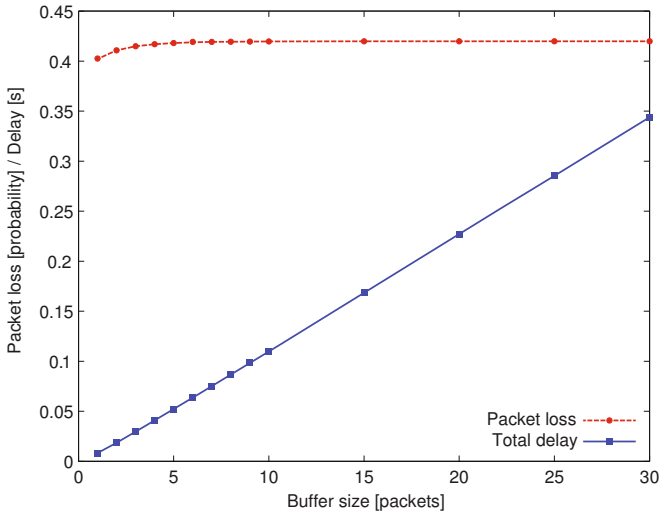


Fig. 3. Delay and loss as buffer size is varied. 140% total offered load.

Let us now examine the case where the network is heavily loaded, the load exceeding the network capacity. In this scenario, buffers become immediately full and the excess load is dropped as the queues cannot keep up with new arrivals. Figure 3 illustrates the performance in such circumstances. In this example, we consider an offered load of 140% of the total capacity, which results in slightly more than 40% of the packets being discarded. Also, as the queues remain full

the total delay is dominated by the queueing delay, which increases linearly with the buffer size.

What is more interesting to observe is the effect of the buffer size on the performance of the network under moderate-to-high loads. For this purpose, we analyse the packet loss rate and delay in a scenario where the total load is 85% of the network capacity. This results are depicted in Figure 4. As expected, since the traffic volume is significant (however, not overloading the network) we see the delay increasing as we increase the buffer size. On the other hand, the packet loss rate initially decreases as we increase the buffer size, but we find that beyond approximately 5 packets of buffering, surprisingly the packet loss rate actually increases as the queue expands.

We can gain some insight into why increasing buffering actually increases losses by considering the same network with $n = 10$ stations, where we fix the queue length of $n - 1$ of them to $K = 5$ and vary the buffer size of one other station. The resulting packet loss rates in this scenario are shown in Figure 5. We see that as we increase the buffer size of one station, the packet loss rate it experiences decreases, while the packet loss rate of the other stations in the network increases. This suggests that, as we increase contention for the wireless medium, by reducing the packet loss of one station, we are decreasing the time available to other stations through a mix of collisions and successful transmissions.

In the following subsection, we demonstrate this effect more clearly by using the simplified $M/M/1/K$ model introduced in Sec. 2.2.

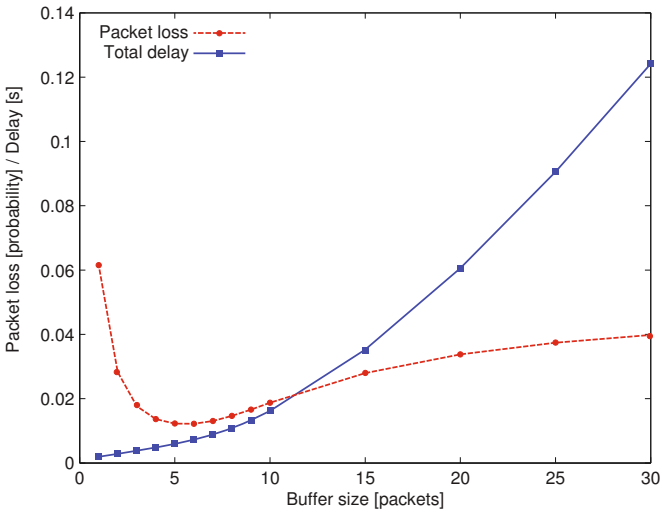


Fig. 4. Delay and packet loss as buffer size is varied. 85% total offered load.

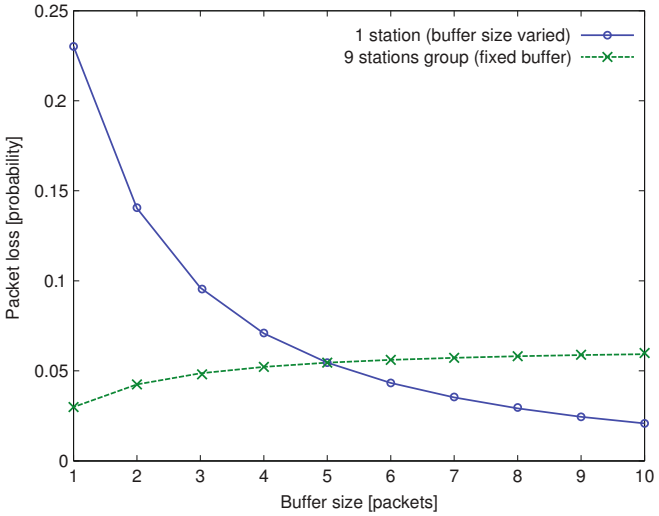


Fig. 5. Packet loss behaviour as buffer size is varied only for one station

3.2 Numerical Results with the M/M/1/K Model

To further illustrate the impact of the buffer size on the packet loss rate at moderate-to-high traffic loads, and confirm that the non-monotonic behaviour observed in our numerical analysis based on the model of Liu et al. is not an artefact of the model, here we consider an equivalent network with $n = 10$ active stations running slotted Aloha and study the packet loss as predicted by our $M/M/1/K$ model.

Although the network parameters are not directly comparable, we consider stations with a maximum transmission probability of $\tau_0 = 0.15$, which is larger than the optimal transmission probability of $1/10$. In this scenario, provided the offered load is large enough, by varying the buffer size we can move from a situation where the actual transmission probability is below the optimal value, to a situation where it exceeds the optimal value. For an offered load of $\lambda = 0.045$ packets per slot per station, the resulting packet loss is shown in Figure 6, where we identify a pattern very similar to the packet loss observed for the 802.11 case depicted in Figure 4.

To add more perspective, we also calculate the transmission probabilities of the stations and compare them to the optimal value of $1/10$. The results are shown in Figure 7. We see that as we increase the buffer size K, the actual transmission probability τ increases, indicating that there are packets available for transmission at the head of the queue more often. However, for the parameters shown, once $K > 4$ the value of τ exceeds the optimal value. As we saw in Figure 6, this results in increased packet loss, since the number of collisions caused by simultaneous transmissions outweighs the extra transmission attempts.

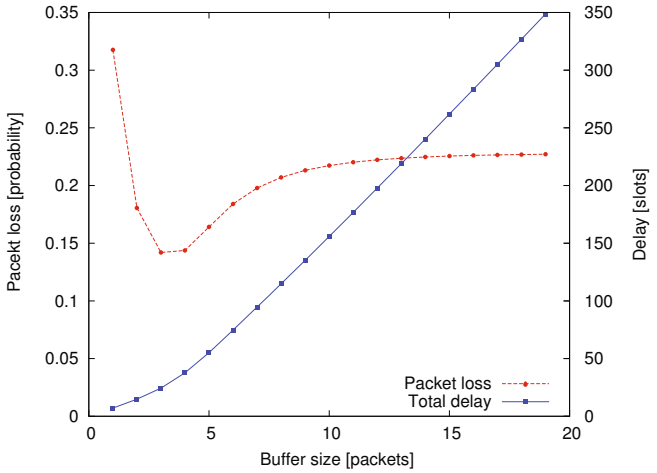


Fig. 6. Packet loss & delay behaviour as buffer size is varied. ALOHA network ($n = 10$).

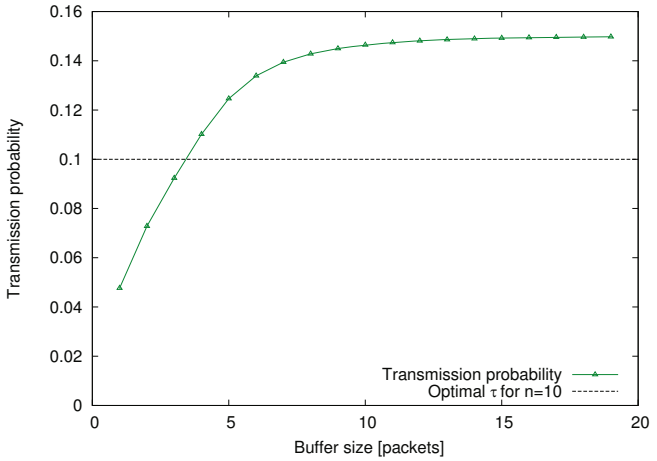


Fig. 7. Transmission probability as buffer size is varied

Following these observations, we believe that the behaviour shown by the 802.11 system in Figure 4 can be understood in the same way. This finding is in fact consistent with earlier work that demonstrates the throughput of an 802.11 network can decrease as the offered load increases even when buffering is fixed [10], as our results indicate that increased buffering may also increase the effective transmission probability for 802.11 beyond its optimal value, and consequently decrease network throughput.

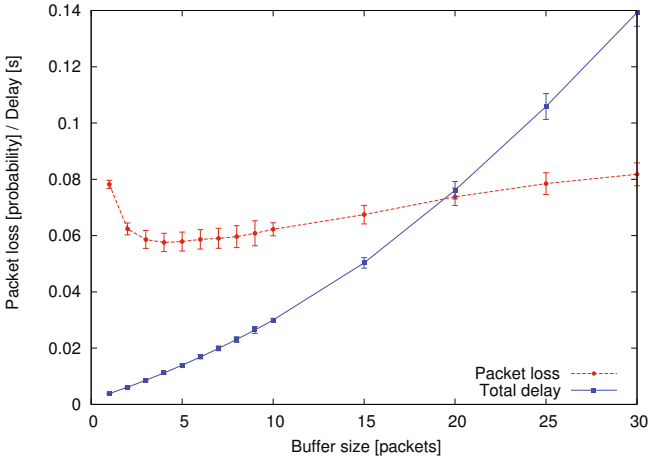


Fig. 8. Performance evaluation of a simulated network with real-time traffic

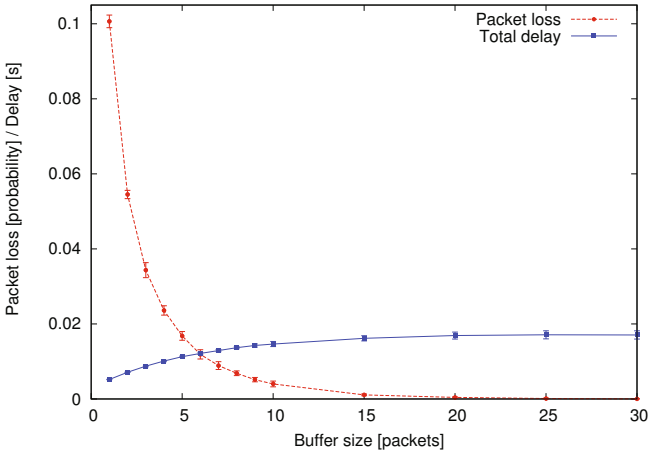


Fig. 9. Performance of an optimally configured network with real-time traffic

3.3 Simulation Results

To conclude our performance analysis, we consider a realistic scenario that is likely to be impacted by the identified non-monotonic behaviour of the packet loss when the buffer size is varied. Specifically, our focus is on interactive applications such as online games, video conferencing or voice over WLAN. Therefore, we run simulation experiments with a 10-node network, where each station runs a delay sensitive application that generates 425 kb/s of traffic, mapped over the voice queue at the MAC layer. For this purpose, we use an event-based simulator that we implemented in C/C++, which follows the 802.11 protocol rules and timing, and consider synthetic constant bit-rate traffic sources, generating 500-byte packets. Given the MAC layer configuration for the access category

employed ($CW_{min} = 8$, $CW_{max} = 16$) and the traffic volume, the network will be operating in a medium-to-high load regime. After a 30 s warm-up period, we measure the average packet loss rate and delay over a duration of 5 min. We repeat the simulations 10 times and plot the average and 95% confidence intervals for the two metrics in Figure 8. We identify the same trend as observed in the previous numerical analysis, which strengthens our belief that appropriate MAC and buffer tuning is necessary to optimise the performance of such applications.

4 Discussion

We have seen that due to the contention-based nature of the 802.11 protocol, packet loss may exhibit a non-monotonic behaviour when the buffer size is increased, as the transmission probability becomes larger than the optimal value. In addition, the total delay increases substantially, which may have a severe impact on interactive applications. On the other hand, such applications are less susceptible to packet loss [5]. In these circumstances it is important to appropriately tune the MAC parameters and buffer size of the stations, to meet latency requirements.

By setting the maximum transmission probability τ_0 (corresponding to saturation) of the nodes to a fixed value, one can ensure that the actual transmission probability will not increase beyond this point irrespective of the offered load. Further, if this is set to the optimal value, delay can be optimised, while the non-monotonic behaviour of the loss identified previously can be eliminated. Thus the total delay can be bounded by appropriate tuning of the buffer size.

Setting τ_0 to a fixed value, is easily achievable by configuring nodes with $CW = CW_{min} = CW_{max}$, i.e. $\tau_0 = 2/(CW + 1)$. For optimality, this can be performed either in a centralised way from the AP, which holds information about the network size (see [3] for a detailed discussion), or in a distributed manner, by only observing the channel conditions and dynamically tuning CW, as in e.g. [11]. With this setting, as long as the offered load does not exceed the service rate, a small buffer would suffice, while as stations become overloaded, delay is directly proportional to the buffer size. To demonstrate this, consider again the earlier simulation experiment, with $n = 10$ nodes sending each 425 kb/s to the AP, but this time contending with the optimal CW configuration. As shown in Figure 9, packet loss drops to zero as the buffer size exceeds 10 packets, while the delay remains bounded below 20 ms, which is perfectly suitable for the real-time traffic considered [4].

5 Conclusions

In this paper we have demonstrated that increasing the buffer size in an 802.11 network may actually increase packet loss rates. We demonstrate this result using a state-of-the-art Markov chain model of buffered 802.11 and a simpler model of a slotted Aloha network with a M/M/1/K buffer, as well as through simulations. We show that this effect can be understood in terms of an increased

number of packets reaching the MAC service, which leads to increased contention and consequently to a throughput drop. This result has implications for the provisioning of buffers for traffic with a delay/packet loss trade-off, such as interactive applications. We show that by appropriately configuring the channel access parameters of the protocol, latency can be controlled by tuning the buffer size according to the application requirements.

References

1. Asmussen, S.: Applied Probability and Queues, 2nd edn. Springer (2003)
2. Bertsekas, D.P., Gallager, R.G.: Data networks, vol. 2. Prentice-Hall (1987)
3. Bianchi, G.: Performance analysis of IEEE 802.11 distributed coordination function. IEEE Journal on Selected Areas in Communications 18(3), 535–547 (2000)
4. Carrig, B., Denieffe, D., Murphy, J.: Supporting first person shooter games in wireless local area networks. In: Proc. IEEE PIMRC, pp. 1–6 (September 2007)
5. Claypool, M., Claypool, K.: Latency and player actions in online games. Commun. ACM 49 (2006)
6. Duffy, K., Ganesh, A.J.: Modeling the impact of buffering on 802.11. IEEE Communications Letters 11(2) (February 2007)
7. Huang, K., Duffy, K.R., Malone, D.: On the validity of IEEE 802.11 MAC modeling hypotheses. IEEE/ACM Transactions on Networking 18(6), 1935–1948 (2010)
8. Little, J., Graves, S.: Little's law. In: Building Intuition. International Series in Operations Research & Management Science, vol. 115, pp. 81–100. Springer US (2008)
9. Liu, R.P., Sutton, G., Collings, I.B.: A new queueing model for QoS analysis of IEEE 802.11 DCF with finite buffer and load. IEEE Transactions on Wireless Communications 9(8), 2664–2675 (2010)
10. Malone, D., Duffy, K., Leith, D.: Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions. IEEE/ACM Transactions on Networking 15(1), 159–172 (2007)
11. Patras, P., Banchs, A., Serrano, P., Azcorra, A.: A Control-Theoretic Approach to Distributed Optimal Configuration of 802.11 WLANs. IEEE Transactions on Mobile Computing 10(6), 897–910 (2011)
12. Zhai, H., Kwon, Y., Fang, Y.: Performance analysis of IEEE 802.11 MAC protocols in wireless LANs. Wireless Communications and Mobile Computing 4(8), 917–931 (2004)

On the Capacity of a PPM UWB Multiple-Access System with a Single User Noncoherent Reception*

Dmitry Osipov[1,2], Alexey Frolov[1], and Victor Zyablov[1]

[1] Institute for Information Transmission Problems
Russian Academy of Sciences,
19 Bolshoy Karetny per. Moscow 127994, Russia
[2] National Research University Higher School of Economics,
20 Myasnitskaya Ulitsa Moscow 101000, Russia

Abstract. We consider an uncoordinated multiple-access system that employs a modulation technique, in which the probability of suppressing the signal sent by a certain user can be considered negligible, to transmit information via a wireless channel (e.g. time hopping (TH) with pulse position modulation (PPM)).This channel can be considered as an A channel (channel without intensity information) [1]. For this channel a new method of transmission is proposed. The expression for the capacity of a multiple access system employing the proposed transmission method (for the single user reception case) is obtained. Both non-asymptotic and asymptotic formulas are derived and asymptotic behavior of the capacity (for the single user reception case) is studied.

1 Introduction

Let us consider a communication scheme which employs Q-ary pulse position modulation (PPM) in a wireless channel (e.g. TH UWB with noncoherent reception [2,3]). In this case the channel consists of Q subchannels which correspond to time slots. To transmit the i^{th} element of an Q-ary alphabet the user needs to transmit energy (e.g., a short pulse) in the i^{th} subchannel. We can say that the Q-ary symbol is transmitted as a binary vector of length Q and weight one. The detector at the receiver measures the energy in the i^{th} subchannel and decides if "1" or "0" was transmitted by comparing the energy with the threshold. In this case the probability of receiving "1" as "0" is much smaller then the probability of receiving "0" as "1" as we need to suppress the energy in the first case. Therefore we can consider an idealized model model of such a channel: a multiple access vector disjunctive channel. This channel (the A channel or channel without intensity information) was introduced in [1]. Let us consider this channel model in more detail. Let us denote the number of active users by S, $S \geq 2$. So for some time τ the channel inputs are binary vectors $\mathbf{x}_i^{(\tau)}$, $i = 1, 2, \ldots, S$ of

* This work has been supported in part by the RFBR grant (N 12-07-31035 mol_a).

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 49–57, 2013.

length Q and weight one, and the channel output at time τ is an elementwise disjunction of vectors at input

$$\mathbf{y}^{(\tau)} = \bigvee_{i=1}^{S} \mathbf{x}_i^{(\tau)}.$$

The capacity of the A channel was investigated in [1,4–6] both in case of coordinated and uncoordinated transmission. In this paper we focus on uncoordinated multiple-access, i.e a type of transmission in which other users are treated as a noise. Uncoordinated multiple-access is preferable for high rate applications where joint decoding is unavailable for complexity reasons. Please note that within the scope of the model under consideration each subchannel of the channel in use is a binary Z channel, the channel in which "1" is always transmitted correctly, and "0" is replaced by "1" with some probability.

We propose a new method of transmission, which is a modification of a method from [7]. In accordance to this method each user is given a subrange of $q \ll Q$ subchannels. Subranges are allocated dynamically. By this means we significantly decrease the probability of collision.

Our contribution is as follows. We obtain the capacity of a single user for the proposed transmission method. Both non-asymptotic and asymptotic formulas are derived. The capacity is increasing when q is growing, it is decreasing with S, enabling however the multiple access system under consideration to provide simultaneous transmission even for a large number of active users. It will be shown that starting from $q = 32$ the obtained asymptotic result is very close to the upper bound from [4], obtained in case of $q = Q$.

2 Transmission and Reception

We consider an uplink channel. All the users transmit information to the base station. Let us recall that the channel consists of Q subchannels. A time interval during which one vector of length Q is transmitted will be called a tact. Assume that each user is given a subrange of q subchannels.

Let us assume that all the users use the same alphabet – symbols of $\mathbb{F}_q$. Let us enumerate the elements of the field $\mathbb{F}_q$ in some order as follows

$$\mathbb{F}_q = \{\, \alpha_1, \alpha_2, \ldots, \alpha_q \,\}.$$

Let us denote by

$$\psi : \mathbb{F}_q \to \{0,1\}^q$$

a mapping that maps every field element $\alpha_i \in \mathbb{F}_q$ to a binary column vector of length q having a single nonzero element at the i^{th} position. The vector positions are counted from 1 to q.

To transmit a symbol $\beta \in \mathbb{F}_q$ a user transmits a vector $\mathbf{v} = \psi(\beta)$ in the subrange given to the user. The subranges are allocated dynamically. For this purpose permutations of length Q are used. Permutations used by a current user are known to nobody except for a "transmitter-receiver" pair.

So the transmission can be seen in such a way. A vector $\mathbf{V} = [\mathbf{v}^T\, 0\, 0 \ldots 0]^T$ of length Q is formed by adding $Q - q$ zeros to the vector $\mathbf{v}$. Before sending the vector $\mathbf{V}$ to the channel, a permutation of its elements is made (a new permutation is used for each vector). In what follows we assume the permutations to be chosen equiprobably and independently from the set of all the $Q!$ possible permutations.

An individual receiver corresponds to each user in the system. For permutations to be known both on the transmitter and the receiver it is advisable to use pseudorandom number generators, which are a part e.g. of any system based on frequency hopping [8].

3 Channel for the i^{th} User

Let us designate a symbol sent by the i^{th} user as $X \in \psi\left(\mathbb{F}_q\right)$, and the output of the channel for the i^{th} user as $Y_Q \in \{0,1\}^Q$(the inverse permutation is assumed to be applied). Mind that in case of uncoordinated multiple-access other users are treated as a noise. The channel for the i^{th} user is shown in Fig. 1.

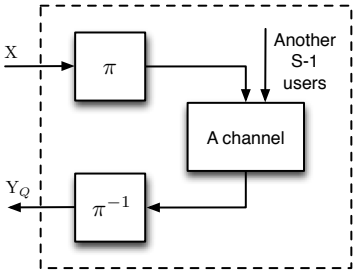


Fig. 1. Channel for the i^{th} user

In what follows we represent Y_Q as $(Y_q\, Y_{Q-q})$, where Y_q corresponds to the subrange given to the i^{th} user and Y_{Q-q} corresponds to the rest of the subchannels. Note that Y_{Q-q} does not depend on X.

Remark 1. As independent and equiprobable permutations are used by other users their 1s are added to Y_Q equiprobably. And thus the channel can be considered as a vector Z channel with transition probability

$$p(1|0) = 1 - \left(1 - \frac{1}{Q}\right)^{S-1}$$

in each subchannel.

Remark 2. It can be seen that the channel under consideration is in fact a symmetric channel.

4 Capacity

Let us determine the capacity C of the channel from Fig. 1. We have

$$C(q, Q, S) = \max\left[I(X; Y_Q)\right],$$

where the maximum is taken over all the distributions at input.

In accordance with the Remark 2 the channel is symmetric and the maximum is reached for the *uniform* distribution at input. In what follows we assume the distribution of X to be uniform.

Lemma 1. *Let the distribution of X be uniform, then the following equality holds*

$$I(X; Y_Q) = I(X; Y_q)$$

Proof. We have

$$I(X; Y_Q) = H(X) - H(X|Y_q, Y_{Q-q}).$$

Thus we only need to prove that

$$H(X|Y_q, Y_{Q-q}) = H(X|Y_q). \tag{1}$$

Let $\mathbf{v}_W$ be a vector of weight W such that $\mathbf{v}_W \wedge \mathbf{x}_i = \mathbf{x}_i$, consider the probability $P(X = \mathbf{x}_i | Y_q = \mathbf{v}_W, Y_{Q-q} = \mathbf{y}_{Q-q})$.

$$\begin{aligned} P(\mathbf{x}_i|\mathbf{v}_W, \mathbf{y}_{Q-q}) &= \frac{P(\mathbf{v}_W, \mathbf{y}_{Q-q}|\mathbf{x}_i)P(\mathbf{x}_i)}{P(\mathbf{v}_W, \mathbf{y}_{Q-q})} \\ &= \frac{P(\mathbf{v}_W, \mathbf{y}_{Q-q}|\mathbf{x}_i)P(\mathbf{x}_i)}{\sum\limits_{j=1}^{q}\left[P(\mathbf{v}_W, \mathbf{y}_{Q-q}|\mathbf{x}_j)P(\mathbf{x}_j)\right]} \\ &= \frac{P(\mathbf{v}_W, \mathbf{y}_{Q-q}|\mathbf{x}_i)}{\sum\limits_{j=1}^{q} P(\mathbf{v}_W, \mathbf{y}_{Q-q}|\mathbf{x}_j)} = \frac{1}{W}. \end{aligned}$$

It can be seen that this probability does not depend on $\mathbf{y}_{Q-q}$, thus

$$\begin{aligned} P(\mathbf{x}_i|\mathbf{v}_W) &= \sum_{\mathbf{y}_{Q-q}}\left[P(\mathbf{x}_i|\mathbf{v}_W, \mathbf{y}_{Q-q})P(\mathbf{y}_{Q-q})\right] \\ &= P(\mathbf{x}_i|\mathbf{v}_W, \mathbf{y}_{Q-q}). \end{aligned}$$

We only need to substitute this result for (1). □

Corollary 1. *Thus it can be seen that Y_{Q-q} adds no information. In what follows we consider only Y_q, i.e.*

$$C(q, Q, S) = I(X; Y_q) = \log_2 q - H(X|Y_q). \tag{2}$$

We need the following lemma. Mind $\mathbf{v}_W$ is a fixed vector of weight W which covers $\mathbf{x}_i$.

Lemma 2. *We have*

$$P(Y_q = \mathbf{v}_W | X = \mathbf{x}_i) = P(W) = \sum_{i=0}^{W-1} \left[(-1)^i \binom{W-1}{i} \left(\frac{Q-q+W-i}{Q} \right)^{S-1} \right].$$

Proof. We need to apply the inclusion-exclusion formula. □

Now we are ready to prove a theorem.

Theorem 1

$$C(q, Q, S) = \log_2 q - \sum_{W=1}^{q} \left[\binom{q-1}{W-1} P(W) \log_2 W \right].$$

Proof. Using the fact $P(\mathbf{x}_i | \mathbf{v}_W) = 1/W$, Lemma 2 and (2) we obtain the result strived at. □

In Fig. 2 and Fig. 3 the dependencies of the capacity C and relative capacity $C/\log_2 q$ on q are shown. Other parameters are chosen in such a way: $Q = 1024$ and $S = 100$. It can be seen that C increases when q grows, at the same time $C/\log_2 q$ becomes smaller when q grows.

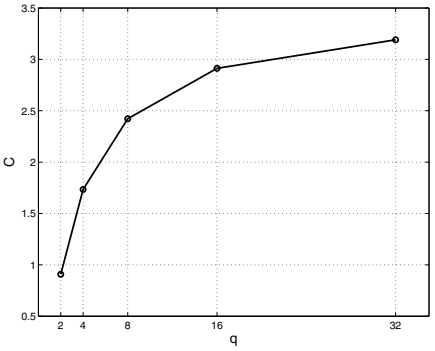


Fig. 2. The dependency of C on q

In Fig. 4 the dependency of C on S is shown for $Q = 1024$ and $q = 16$. The capacity decreases with S, but at the same time the system enables the work of a very large number of users, i.e. $C(q = 16, Q = 1024, S = 2000) = 0.2$.

Let us introduce a notion of a sum capacity.

$$C_{Sum}(q, Q, S) = SC(q, Q, S).$$

In Fig. 5 the dependency of a sum capacity per subchannel C_{Sum}/Q on S is shown for $Q = 1024$ and $q = 16$. One can see that there is a maximum of C_{Sum}/Q at some S.

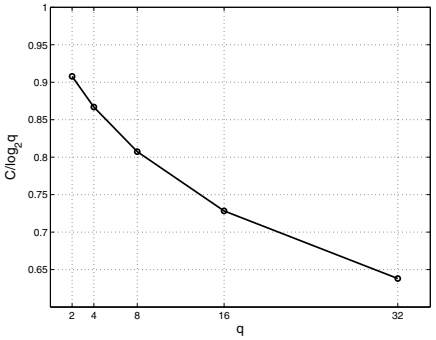


Fig. 3. The dependency of $C/\log_2 q$ on q

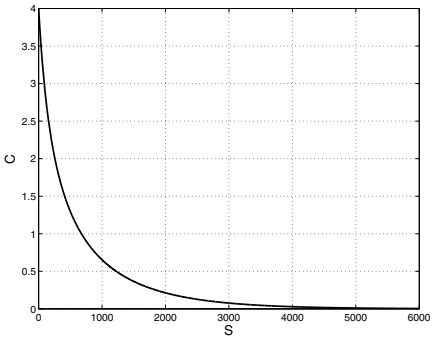


Fig. 4. The dependency of C on S

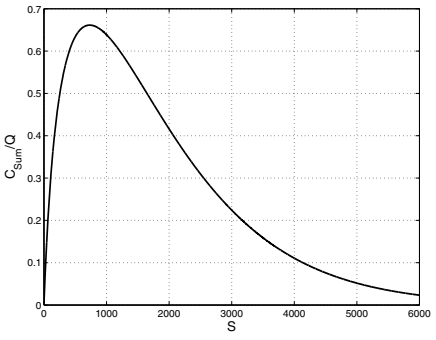


Fig. 5. The dependency of C_{Sum}/Q on S

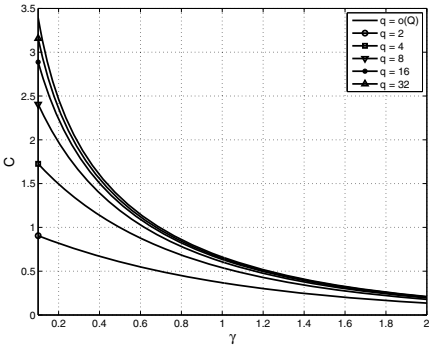


Fig. 6. The dependency of C on γ

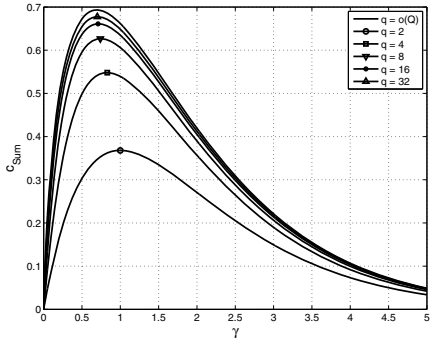


Fig. 7. The dependency of C_{Sum} on γ

Now let us consider the asymptotic ($Q \to \infty$) capacity. Let $S = \gamma Q$. Let us introduce the notions of

$$C(q, \gamma) = \lim_{Q \to \infty} C(q, Q, \gamma Q).$$

and

$$C_{Sum}(q, \gamma) = \gamma \lim_{Q \to \infty} C(q, Q, \gamma Q).$$

Theorem 2. *The following statements hold*

1. *Let q be fixed, then*

$$C(q, \gamma) = \sum_{W=1}^{q} \left[\binom{q-1}{W-1} e^{-\gamma(q-W)} (1 - e^{-\gamma})^{W-1} \log_2 \left(\frac{q}{W} \right) \right];$$

2. *Let $q \to \infty$ when $Q \to \infty$, then*

$$C(\gamma) = \lim_{q \to \infty} C(q, \gamma) = -\log_2(1 - e^{-\gamma}).$$

Proof. To prove the first statement it is sufficient to mention that

$$\lim_{Q\to\infty} P(W) = e^{-\gamma(q-W)}(1-e^{-\gamma})^{W-1}.$$

To prove the second statement we use the method from [4]. Please, for more details see [4], for the sake of brevity we are omitting the proof here. □

Remark 3. Note, that $C(\gamma)$ coincides with the bound from [4] obtained for $q = Q$. So one can obtain the same result with smaller values of q, e.g. $q = o(Q)$.

In Fig. 6 and Fig. 7 the dependencies of $C(q,\gamma)$ and $C_{Sum}(q,\gamma)$ are shown. We see that already for $q = 32$ $C(q,\gamma)$ and $C_{Sum}(q,\gamma)$ are very close to $C(\gamma)$ and $C_{Sum}(\gamma)$. So there is no need to use large values of q and thus we reduce the probability of collision significantly. We can also significantly reduce the complexity of decoding in the signal-code construction from [7].

5 Conclusion

Hereinabove an idealized model of the multiple access system employing Q-ary PPM in a wireless channel is considered. A new transmission method has been proposed: within the scope of this method each user is given a subrange of $q \ll Q$ subchannels (time slots). Subranges are allocated dynamically. We obtain the expression for the capacity of a multiple access system employing the proposed transmission method (for the single user reception case). Both non-asymptotic and asymptotic formulas are derived. The capacity is increasing when q is growing, it is decreasing with S, enabling however the multiple access system under consideration to provide simultaneous transmission even for a large number of active users. Hereinabove it has been demonstrated that starting from $q = 32$ the obtained asymptotic result is very close to the upper bound from [4], obtained for the case of $q = Q$. Thus, there is no need to use large values of q and therefore the probability of collision can be significantly reduced.

References

1. Chang, S.-C., Wolf, J.: On the t-User m-Frequency Noiseless Multiple-Access Channel with and without Intensity Information. IEEE Transactions on Information Theory, 41–48 (January 1981)
2. Tsang, T., El-Gamal, M.: Ultra-Wideband (UWB) Communications Systems: An Overview. In: 3rd International IEEE-NEWCAS Conference, pp. 381–386 (June 2005)
3. Witrisal, K., Leus, G., Janssen, G., Pausini, M., Troesch, F., Zasowski, T., Romme, J.: Noncoherent Ultra-Wideband Systems. IEEE Signal Processessing Magazine, 48–66 (July 2009)
4. Wilhelmsson, L., Zigangirov, K.Sh.: On the Asymptotic Capacity of a Multiple-Access Channel. Problems of Information Transmission, 12–20 (1997)

5. Bassalygo, L.A., Pinsker, M.S.: Evaluation of the Asymptotics of the Summarized Capacity of an m-Frequency t-User Noiseless Multiple-Access Channel. Problems of Information Transmission, 3–9 (2000)
6. Han Vinck, A., Keuning, K.: On the Capacity of the Asynchronous t-User m-Frequency Noiseless Multiple-Access Channel without Intensity Information. IEEE Transactions on Information Theory, 2235–2238 (November 1996)
7. Osipov, D., Frolov, A., Zyablov, V.: Multiple Access System for a Vector Disjunctive Channel. Problems of Information Transmission 48(3), 243–249 (2012)
8. Zigangirov, K.Sh: Theory of Code Division Multiple Access Communication. IEEE Press (2004)

Detecting Transmission Power Misbehaviour in Wi-Fi Networks

Szymon Szott, Marek Sikora, Marek Natkaniec, and Krzysztof Loziak

AGH University of Science and Technology
Faculty of Computer Science, Electronics and Telecommunications
al. A. Mickiewicza 30, 30-059 Krakow, Poland
{szott,msikora,natkaniec,kloziak}@kt.agh.edu.pl

Abstract. In Wi-Fi networks, transmission (TX) power levels are constrained by regulatory limits. However, the emergence of flexible MAC drivers allows the easy modification of PHY and MAC layer parameters. This has enabled users to attempt to violate these limits. Such actions, which we refer to as TX power misbehaviour, allow users to achieve higher throughput due to the capture effect. Detecting this type of misbehaviour is challenging because of the inability to directly measure the TX power of a user's device. Therefore, in this paper, we propose detection methods for (a) determining if the TX power violates regulatory limits and (b) estimating the TX power value. Separate methods are proposed for single and multi-rate modes of operation. Simulation results verify that the proposed methods may be successfully used to detect TX power misbehaviour. Therefore, they can be implemented in a general misbehaviour detection and reaction architecture, also presented in this paper.

Keywords: Detection, EIRP, misbehaviour, transmission power, Wi-Fi.

1 Introduction

Wireless Fidelity (Wi-Fi) networks based on the IEEE 802.11 standard [1] have become one of the most popular means of Internet access, due to their low cost, high transmission speed, license-free operation, and constant improvement through new amendments [2]. However, Wi-Fi networks are prone to misbehaviour for two reasons. First, the IEEE 802.11 standard does not contain any incentives for stations to conform to the specification [3]. Second, we are observing the emergence of new flexible MAC solutions, such as the Wireless MAC Processor [4], which allow Wi-Fi devices to rapidly adapt to evolving contexts and service needs.

One important case of misbehaviour is setting high transmission (TX) power values. Each country imposes certain regulatory limits on the Equivalent Isotropic Radiated Power (EIRP) of a device (e.g., 100 mW is a common limit in Europe). Such limits are often embedded in wireless MAC drivers. However, for the two reasons stated above, a Wi-Fi device may be configured to use TX power values which violate regulatory limits. The main advantage of such behaviour is an increased probability of

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 58–68, 2013.

gaining medium access because of the capture effect [5]. Additionally, a misbehaving user may expect to achieve an extended range of operation.

The main problem with TX power misbehaviour is that it is difficult to detect. First, the EIRP cannot be measured directly at the offending device. Therefore, indirect measures need to be used. Second, the distance of the transmitting stations from the receiver is unknown. This can also highly influence the detection criteria. With these obstacles in mind, within the FLAVIA project [6], we have attempted to nonetheless perform the detection of TX power misbehaviour. To achieve this goal, we use a dedicated misbehaviour detection architecture, developed within the project, for which we have designed algorithms to detect this type of misbehaviour based on radio measurements. Simulation results verify that the proposed approach is promising and open to further extensions.

The remainder of this paper is organized as follows. In Section 2 we provide a brief state of the art. In Section 3, we describe our misbehaviour detection architecture. General issues regarding detecting abnormal TX power are considered in Section 4. In Section 5, we provide detailed analyses for two special cases: single-rate and multi-rate networks. Finally, Section 6 concludes the paper and discusses future work.

2 State of the Art

Most research work in the area of detecting misbehaviour in IEEE 802.11 has been focused on backoff misbehaviour, which includes selecting a smaller backoff, assigning a fixed backoff or not doubling the contention window (CW) values. Most approaches are based, first, on monitoring the network and recording the observed backoff values of a station and, second, determining whether they are standard compliant. Observations are hindered by such factors as: interference from other transmissions, unsynchronized clocks, and non-deterministic medium access. Therefore, a monitoring service is required to supply accurate and timely information. Having performed the measurements, it is necessary to determine how to classify the station behaviour. For this case the chi-square test has exhibited satisfactory performance [7]. A comparison of other backoff detection schemes can be found in [8] while a general overview of the challenges and solutions related to MAC misbehaviour in Wi-Fi networks can be found in [9].

Detecting high TX power misbehaviour is also related to anomaly detection in Wi-Fi networks. The authors of [10] detect jamming attacks based on the statistical characteristics of the observed signal-to-noise ratio (SNR) values. MOJO, a distributed physical layer anomaly detection system for 802.11 WLANs, is presented in [13]. This system is designed to detect hidden terminals [11, 12], noise and signal strength variations, as well as occurrences of the capture effect. However, to the best of our knowledge, the problem of detecting TX power misbehaviour has not been considered in the literature.

3 Detection Architecture

To counter the risks of misbehaviour in Wi-Fi networks, we have developed, within the FLAVIA project, an architecture for detecting various types of misbehaviour. The architecture consists of several interworking modules (Fig. 1), including the WMP [4]. These modules jointly operate to provide a set of passive monitoring services able to measure several parameters related to radio channel conditions, estimate PHY and MAC parameters based on these measurements, use these results to detect cheating (including high TX power settings), and, finally, deal with stations with misconfigured 802.11 parameters. The execution of misbehaviour reaction rules is realised by the WMP, thanks to its flexible nature. This architecture is expected to be implemented in an access point (AP) and used, e.g., in hotspot scenarios. Throughout the paper we refer to the AP as the station performing misbehaviour detection, although nothing prohibits an ordinary Wi-Fi station from using this architecture.

The Misbehaviour Detection and Reaction (MDR) module is the most important part of the architecture. It is responsible for handling the misconfiguration of IEEE 802.11 parameters. Its operation is based on network measurements obtained from the monitoring module (MONI).

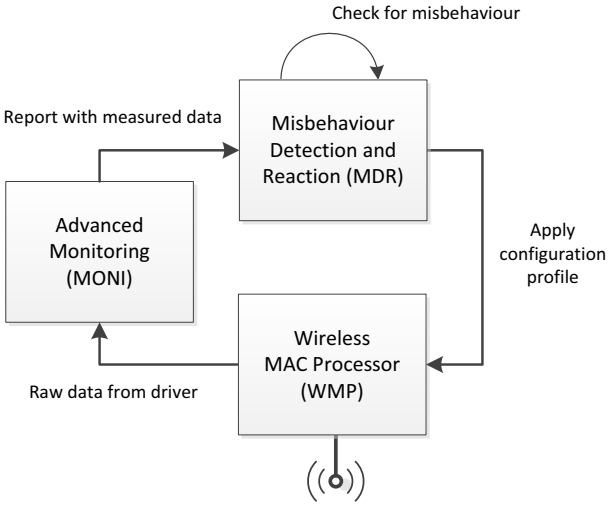


Fig. 1. General overview of detection architecture

MONI is a general monitoring service which adopts a passive approach. This allows the use of regular data transmission interfaces for measurements. MONI works on a frame level – this means that all frames sent and received by each network interface can be examined by the MONI functions. MONI can measure a multitude of parameters. Among the most important, relevant to detecting TX power misbehaviour are the following: supported rates, preamble type, SNR, operation mode, number of retransmission, frame type, sender/receiver MAC address, access category, frame length, correctness of received frame, technology independent SNR, and technology

dependent RSSI (Received Signal Strength Indicator). MONI can also calculate different parameters and statistics, e.g., number of active stations in the neighbourhood, number of received frames, number of transmitted frames, Frame Error Rate (FER), and Bit Error Rate (BER). All these measurements and calculations are performed separately for each network interface of the station.

Based on these measurements, obtained from MONI in the form of periodic reports, MDR detects misbehaving stations and selects methods to encourage such stations to cooperate. The detection procedures for TX power misbehaviour are described in the subsequent sections. The reaction methods can take multiple forms. An offending station may be disassociated from the network or its quality of service may be degraded, e.g., by selectively dropping ACK frames [14]. These reaction methods are applied by configuring the medium access control function in the WMP.

4 Overview of TX Power Detection

The maximum EIRP values, as regulated by law, are defined as the maximum received power value obtainable for a reference 10-meter long link consisting of a transmitter operating with the maximum allowable TX power and a reference receiver equipped with a half wave dipole antenna. In practice, detecting stations exceeding the TX power limit is a complex task because it is in most cases infeasible to perform such a reference measurement. Therefore, we need to infer the TX power from other parameters.

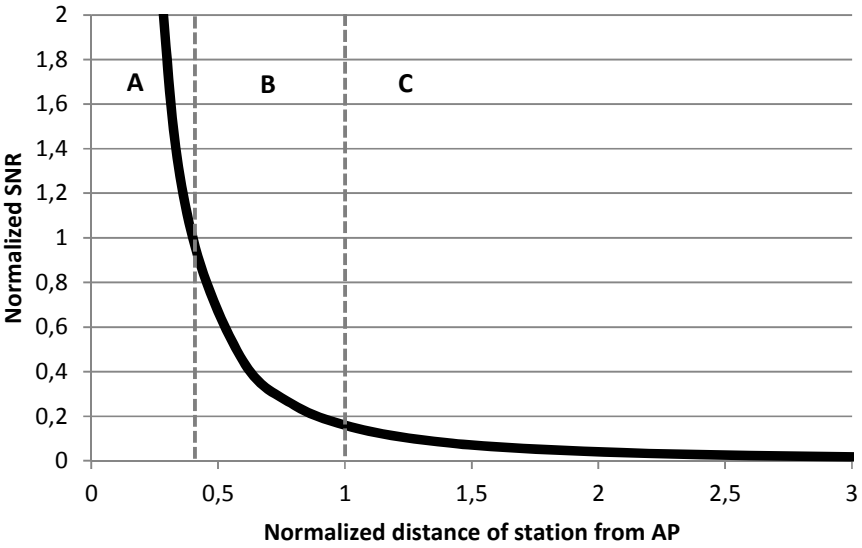


Fig. 2. Three cases of received SNR: A – SNR above regulatory limits, B – SNR within regulatory limits, C – loss of two-way connectivity

One option is to observe the power of received frames from an offending station. Wireless drivers provide information on the SNR of such frames. Fig. 2 illustrates the theoretical change of the SNR of frames received at an AP with the distance of the misbehaving station from the AP. This simplified analysis is presented to show the three cases of received SNR. To this end, we have assumed that SNR follows the inverse-square law. The distance is normalized according to the range of the AP, while the received SNR – according to the allowed limit.

First, if the misbehaving station is close (case A), a direct analysis may be performed. An average SNR exceeding a threshold value defines the obvious case of abnormal power detection. Second, once the station moves further away (case B) the average SNR power stays within the limit of allowable values making detection more challenging. We discuss this case in detail in Section 5. Finally, the station may move out of the range of the AP (case C) and two-way connectivity is lost (though frames from the station may still arrive at the AP, due to the station's higher TX power).

There exist other cases of abnormally increased TX power that cannot be detected. An example would be a station placed very far from the AP, having configured increased TX power and using a highly directive antenna. In this case, the station's EIRP value violates regulatory limits; yet, from the network point of view, the station exhibits no misbehaviour. Therefore, we consider only cases when the increased TX power influences the performance of the network. Furthermore, note that additional measures may need to be taken into account if the two stations use antennas with different gains or the channel is not perfectly symmetric. Such scenarios are out of the scope of this paper.

5 Detailed Detection Methods

If the average SNR of received frames is within regulatory limits, further action needs to be undertaken when attempting to detect TX power misbehaviour. To facilitate the detection process, measurement values taken from the MONI module (Section 3) are used. The exact approach differs depending on the misbehaving station's operation mode. If the station operates in single-rate mode, the difference in the FER of frames exchanged between the AP and the misbehaving station is examined. Otherwise, if the misbehaving station uses a rate adaptation algorithm, the detection is based on differences in data rate usage statistics recorded during communication between the AP and the station under study.

For both single and multi-rate operation, an active data exchange between the detecting and misbehaving stations is required. Additionally, for the detection process to operate correctly, we assume that the TX power value of the AP is set to the maximum power allowed by regulatory limits. Fig. 3 illustrates the detection algorithm, while Sections 5.1 and 5.2 provide the details for the single and multi-rate cases, respectively.

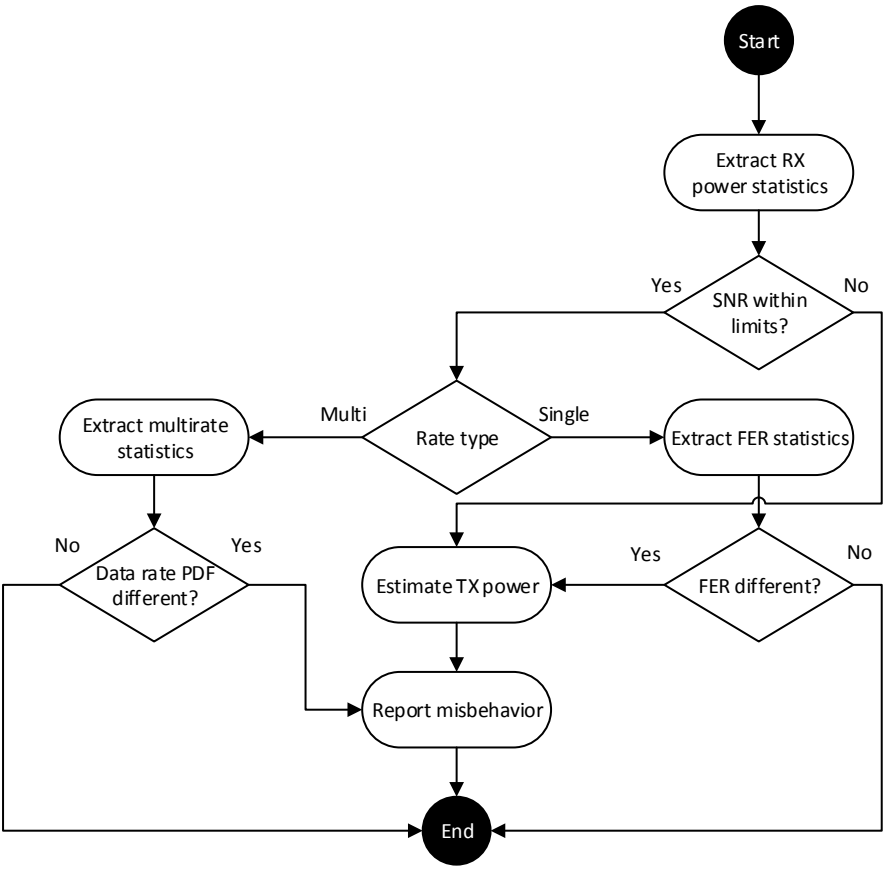


Fig. 3. High TX power detection algorithm

5.1 Single-Rate Operation

For the single-rate case, a station using abnormally increased power can be detected by examining the difference in the FER measured during a data exchange between it and the AP. Due to the high TX power a misbehaving station can achieve a higher SNR which, assuming a symmetric link budget of the radio communication channel and a similar noise level at the receiver input of both stations, results in a different FER. Additionally, thanks to the capture effect a misbehaving station experiences a lower collision probability resulting in a lower number of retransmissions and thus increased data throughput and a further decrease of the FER.

The FER_{STA} experienced by the misbehaving station can be directly estimated by a direct evaluation of the frame check sequence (FCS). The estimation of the FER_{STA} value is therefore straightforward; however, the estimation of the FER_{AP} experienced by the test station requires more effort.

There can be three causes of a frame retransmission in an IEEE 802.11 network. The frame can be retransmitted due to errors introduced by the communication channel, collisions of frames, or, finally, by errors in the transmission of ACK frames. Simulations[1] conducted for various single-rate IEEE 802.11 networks shown that the probability of corruption of the ACK frame are negligible, which means that the FER_{AP} experienced by the AP can be determined based on the reception of ACK frames.

Consider the situation presented in Fig. 4 with a network of two stations, a misbehaving station STA and the AP. Both stations work in a single-rate mode with a nominal data rate of 54 Mb/s and the misbehaving station exceeds the maximum allowed TX power limit by 10 dB.

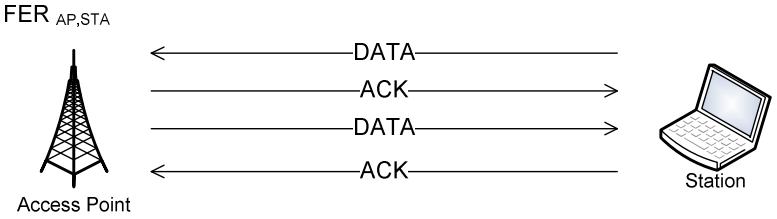


Fig. 4. Detection of high TX Power values

Fig. 5 presents the simulation results of the FER determined using the aforementioned method as a function of distance between stations. The difference in FERs signifies that the station uses a high TX power. However, the efficiency of the proposed method decreases for cases when both the detecting and the misbehaving station experience very low FERs.

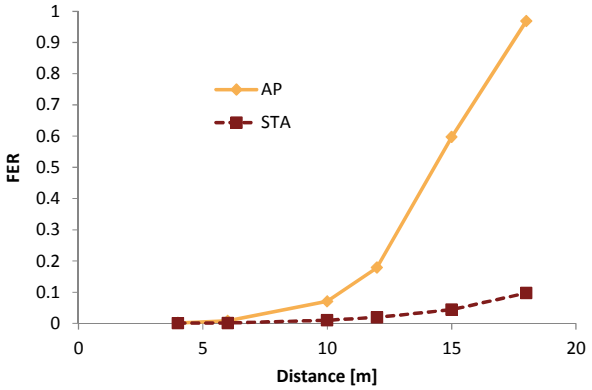


Fig. 5. Difference in FER between the AP and misbehaving station

[1] All simulations were performed using an extended version of the ns-2 simulator.

After detecting the misbehaving station and as a result of knowing the values of FER_{STA} and FER_{AP}, the TX power of the misbehaving station can be estimated. Assuming that the transmission channel exhibits Rayleigh fading, the FER for a particular data rate can be expressed as:

$$FER_{Rayl}(\bar{C}, N, n_{bits}) = \int_0^{+\infty} \frac{1}{\bar{C}} exp\left(\frac{-C}{\bar{C}}\right) FER_{AWGN}(C, N, n_{bits}) dC \quad (1)$$

where $\bar{C}$ is the mean RSSI value at the receiver input, N is the noise at the receiver input and n_{bits} represents the size of the transmitted frame.

Inverting function (1), knowing the size of the transmitted frames and assuming a noise level of -130 dBm, the transmission signal level $C_{STA} = FER_{STA}^{-1}(\bar{C}, N, n_{bits})$ of the misbehaving station and $C_{AP} = FER_{AP}^{-1}(\bar{C}, N, n_{bits})$ of the AP can be calculated. Knowing the TX power of the AP (P_{AP}), the TX power of the misbehaving station can be expressed as $P_{STA} = P_{AP} + \Delta$, where Δ is the difference in power levels and $\Delta = C_{STA} - C_{AP}$.

The presented abnormal TX power estimation method requires significant computational resources and its usage is limited to a certain range of the SNR. Fig. 6 presents simulation results of the difference in power levels Δ as a function of the distance between the detecting and misbehaving station.

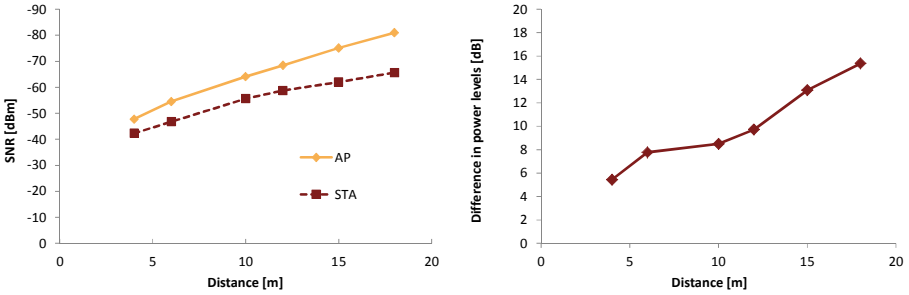


Fig. 6. Determining the difference in power levels

5.2 Multi-rate Operation

Detection of abnormally high TX power in multi-rate IEEE 802.11 networks where stations select the optimum data rate adaptively requires interaction between the AP and the misbehaving station. In multi-rate networks this detection is based on the analysis of the differences between the probability density function of the usage of data rates in the communication between the two Wi-Fi stations. Fig. 6 presents the results of a simulation of an IEEE 802.11multi-rate network where all the stations select data rates using either the ARF (Auto Rate Fallback) [15] or the RBAR (Receiver-Based AutoRate) [15] algorithms. As in previous experiments, the misbehaving station exceeded the maximum allowable power by 10 dB. It can be observed that,

similarly to the single-rate case, the misbehaving station is able to achieve a higher SNR than the AP. This leads to a higher probability of selecting data rates offering higher data throughput at the expense of lower error resilience.

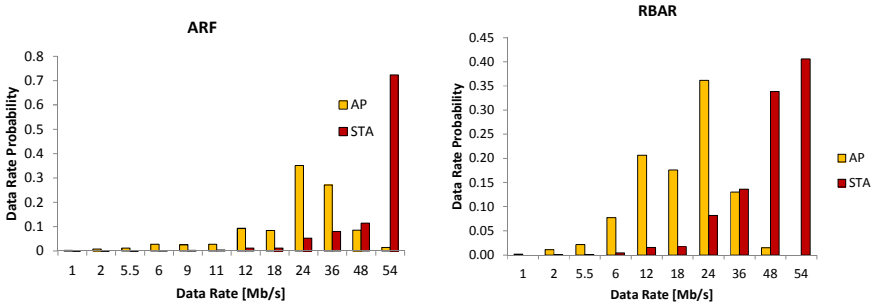


Fig. 7. Detection of TX power misbehaviour in multi-rate networks

Unfortunately, the TX power level of the misbehaving station cannot be determined using equation (1) because the FER is controlled by the data rate selection algorithm, which tries to minimize the FER by selecting the most appropriate data rate for the current propagation conditions. Table 1 presents the result of applying the aforementioned equations to the simulated multi-rate networks using either the ARF or RBAR algorithms for the selection of an optimal data rate. In multi-rate networks the data rate is selected according to the estimation of the propagation conditions.

Table 1. Multi-rate networks using ARF and RBAR. Frame error rates and difference in TX power estimated using (1) and (2).

Data Rate [Mb/s]	ARF			RBAR		
	FER_{AP}	FER_{STA}	Δ [dB]	FER_{AP}	FER_{STA}	Δ [dB]
1	0.021	0.065	-5	0.01147	-	-
2	0.04	0.072	-3	0.00123	0	43
5.5	0.142	0.093	2	0.00417	0.0034	1
6	0.071	0.059	1	0	0	0
9	0.212	0.115	3	-	-	-
11	0.165	0.121	1	-	-	-
12	0.053	0.056	0	0	0	0
18	0.238	0.191	0	1.9E-05	0	61
24	0.046	0.038	1	0	0	0
36	0.269	0.146	3	5.3E-05	2.5E-05	3
48	0.48	0.14	7	0.00139	0.00023	8
54	0.555	0.036	14	-	0	-

Analysing results obtained for a network using ARF one can observe that for data rates from 2 Mb/s to 48 Mb/s which are properly selected by the data rate selection algorithm the frame error value remains unchanged in both directions. Since the ARF is a simple heuristic algorithm using a trial and error mechanism to select an optimal data rate it cannot properly control the 1 Mb/s and 54 Mb/s mode where a higher difference in power levels is noticed. Different results were observed for the RBAR algorithm which selects data rate according to a fast and more precise estimation of channel parameters. The RBAR algorithm is able to maintain a FER for most transmission modes at a very low level independent of the difference in the TX power.

6 Summary

In this paper, we have analysed the problem of TX power misbehaviour, i.e., configuring an IEEE 802.11 device with TX power values exceeding regulatory limits. We have described an architecture for detecting and reacting to misbehaviour. This architecture allows identifying different types of misbehaviour based on passive network measurements. Furthermore, we have discussed general issues related to detecting TX power misbehaviour and then provided methods for both determining such misbehaviour as well as evaluating the TX power of the misbehaving station. The simulation results provided in this validate that the proposed detection methods can be used in IEEE 802.11 APs. As future work, we plan to implement the proposed algorithms in the MDR module, which currently only detects SNR values exceeding a given threshold (Fig. 8). Then, we plan on verifying the simulation analysis using a live testbed. We will consider different values of TX power exceeding the limit, to show the sensitivity of the proposed solution to this very relevant factor.

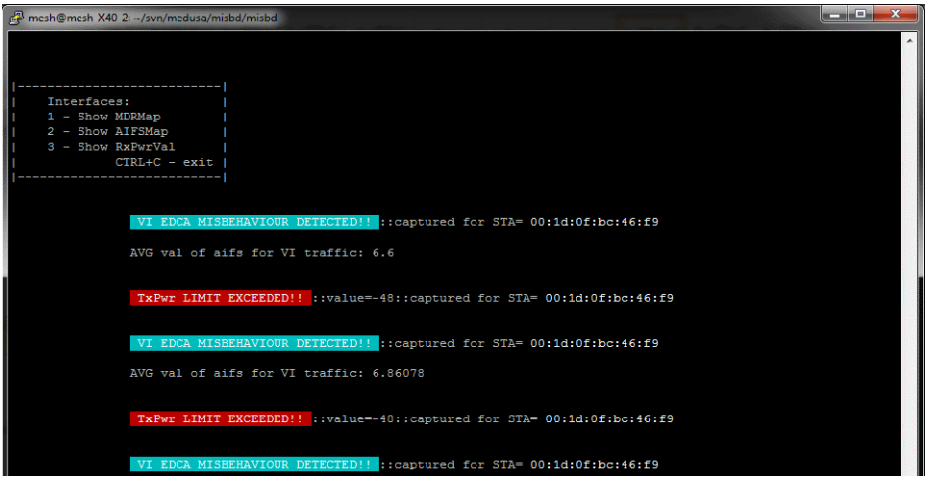


Fig. 8. Output from MDR module showcasing the detection of high TX power as well as back-off misbehaviour in EDCA

Acknowledgements. This work has been supported partially by the European Community's Seventh Framework Programme (FP7-ICT-2009-5) under grant agreement n. 257263 (FLAVIA project) and partially by the AGH University of Science and Technology under contract no. 11.11.230.018.

References

1. IEEE Std 802.11-2012, 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 (March 2012)
2. Kosek-Szott, K., Krasilov, A., Lyakhov, A., Natkaniec, M., Safonov, A., Szott, S., Tinnirello, I.: What's New for QoS in IEEE 802.11? To appear in IEEE Network
3. Szott, S., Natkaniec, M., Pach, A.R.: An IEEE 802.11 EDCA Model with Support for Analysing Networks with Misbehaving Nodes. EURASIP Journal on Wireless Communications and Networking 2010, Article ID 209895, 13 pages (2010), doi: 10.1155/2010/209895
4. Tinnirello, I., Bianchi, G., Gallo, P., Garlisi, D., Giuliano, F., Gringoli, F.: Wireless MAC Processors: Programming MAC Protocols on Commodity Hardware. In: Proc. of IEEE INFOCOM (March 2012)
5. Lee, J., Kim, W., Lee, S.-J., Jo, D., Ryu, J., Kwon, T., Choi, Y.: An experimental study on the capture effect in 802.11a networks. In: Proc. of WinTECH 2007 (2007)
6. FP7-ICT project: FLexible Architecture for Virtualizable future wireless Internet Access—FLAVIA, `http://www.ict-flavia.eu`
7. Szott, S., Natkaniec, M., Canonico, R.: Detecting backoff misbehaviour in IEEE 802.11 EDCA. Wiley European Transactions on Telecommunications 22(1), 31–34 (2011), doi:10.1002/ett.1459
8. Cardenas, A., Radosavac, S., Baras, J.: Performance Comparison of Detection Schemes for MAC Layer Misbehavior. In: Proc. of INFOCOM (2007)
9. Guang, L., Assi, C., Benslimane, A.: MAC layer misbehavior in wireless networks: challenges and solutions. IEEE Wireless Communications 15, 6–14 (2008)
10. Fragkiadakis, A., Siris, V., Petroulakis, N.: Anomaly-based intrusion detection algorithms for wireless networks. Wired/Wireless Internet Communications, 192–203 (2010)
11. Natkaniec, M., Pach, A.R.: A performance analysis of IEEE 802.11 networks in the presence of hidden stations. In: Proc. of the IFIP TC6, WG6.8 5th Workshop on Personal Wireless Communication (PWC 2000), pp. 157–168 (2000)
12. Kosek, K., Natkaniec, M., Vollero, M.L., Pach, A.R.: An Analysis of Star Topology IEEE 802.11e Networks in the Presence of Hidden Nodes. In: ICOIN 2008, Busan, Korea, January 23-25 (2008)
13. Sheth, A., Doerr, C., Grunwald, D., Han, R., Sicker, D.: MOJO: A distributed physical layer anomaly detection system for 802.11 WLANs. In: Proceedings of the 4th International Conference on Mobile Systems, Applications and Services, pp. 191–204. ACM (June 2006)
14. Szott, S., Natkaniec, M., Pach, A.R.: Improving QoS and security in wireless ad hoc networks by mitigating the impact of selfish behaviors: a game-theoretic approach. Wiley Security and Communication Networks 6(4), 509–522 (2013)
15. Kamerman, A., Monteban, L.: WaveLAN-II: A highperformance wireless LAN for the unlicensed band: Wireless. Bell Labs Technical Journal 2(3), 118–133 (1997)

P-Persistent Queue Management to Overcome Channel Failures in IEEE 802.11 Networks for Real-Time Multimedia Streaming

Andrey Guschin, Evgeny Khorov, Anton Kiryanov,
Andrey Lyakhov, and Alexander Safonov

Institute for Information Transmission Problems of Russian Academy of Sciences
(IITP RAS), Bolshoy Karetny per. 19, Moscow, 127994, Russia
{guschin,khorov,kiryanov,lyakhov,safa}@iitp.ru

Abstract. In the paper, we consider real-time video streaming over an IEEE 802.11 network in the presence of short-term channel failures. The failures can dramatically decrease the received video quality at all network stations, even if the channel fails for one station only. In order to assist real-time video traffic to overcome short-term channel failures and to minimize video quality damage, we present the *p*-persistent queue management policy and propose a strategy for adaptive tuning the values of probabilities p assigned to queued packets.

Keywords: *p*-persistent, queue management, video, IEEE 802.11.

1 Introduction

The concept of telecommunication networks has changed greatly in the last few years. The mission of today's networks is not just moving data but providing services to end users. Analysts agree that real time video streaming, IPTV or video conferencing, will prevail in the near future. According to the recent Alcatel report [1], in the United States alone, video traffic is expected to grow 12 times by 2020. Being dominating last mile technologies, wireless networks require deep analysis whether they are able to transmit real-time video.

Transmission of real-time video is more complicated than background traffic, e.g., ftp or http traffic, since it is delay sensitive and, even tolerant to small losses, requires high packet delivery ratio. Meeting such QoS requirements is especially complicated in Wi-Fi networks, working in unlicensed bands.

Most of related papers aim to improve robustness of real-time video streaming in Wi-Fi networks in the case when the channel capacity is insufficient to transmit the whole video stream. They assume that the channel quality does not change with time or the changes are very rare. However, in some cases, we are faced with short-term channel failures during which it is almost impossible to transmit data. These failures are caused by channel or rate switching, short-term interference from home appliances, etc. Different methods and algorithms maintaining video quality in case when channel properties vary gradually, fail to cope

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 69–79, 2013.

with short-term channel failures, which sometimes can dramatically decrease the received video quality.

Apart from that, when an access point is streaming videos to several client stations (STAs), the channel failures which happen to one of the STAs also worsen transmission conditions for other STAs, since channel failures increase the number of transmission attempts and result in more channel resource consumption, decreasing available channel capacity. So, even when the channel fails for a STA, all the STAs may receive distorted video.

In this work-in-progress paper, we address this scenario and develop a method which does the best to improve video quality for STAs with bad channel, while avoiding video quality degradation for STAs with good channel.

The rest of the paper is organized as follows. In Section 2, we analyze previous works related to real-time video streaming. Section 3 states the problem, and Section 4 describes the proposed method to overcome short-channel failures. In Section 5, we describe simulation environment and experiments, which help us to evaluate performance of the proposed method and show obtained numerical results. Finally, Section 6 concludes the paper.

2 Related Works

At our best knowledge, almost all papers aiming to improve robustness of video transmission in Wi-Fi networks consider the case when the channel properties vary slightly and remain constant for a long period of time. In this case, video packets may be dropped for two causes [2]:

- Due to high packet error rate (PER), all transmission attempts appear to be unsuccessful.
- The packet is silently discard since the size of the packet queue reaches predefined threshold.

The problems are interconnected. Increasing the retry limit to cope with high packet error rate, we consume more channel resources and decrease the available channel capacity, which, in turn, leads to the growth ot the queue size. So, the key idea of most methods is to forcedly drop the least important packets from the queue when the traffic load exceeds the estimated channel capacity and it is impossible to transmit all packets. The appliance of these methods is explained by the complex structure of video streams where various packets are of different importance, i.e. the loss of a packet may lead to negligible video distortion while the loss of another packet may result in long-term artifacts.

Based on this idea, the I-Frame Delay (IFD) method [3], may be the most widely-known method, takes into consideration that MPEG4 and H.264 video frames may be of 3 types: I (the most valuable), P and B (the less valuable). The IFD method defines a queue size threshold and operates only when this threshold is reached. When a new frame is enqueued, the method selects a frame to be dropped by taking into account frames importance. It may be the new frame or a frame from the queue. Although the authors describe the method for

the threshold equal to one, they mention that it may be higher [4]. In spite of it is not apparent how the IFD method operates when several video streams are transmitted, it is absolutely clear that the method does not improve transmission reliability of not dropped frames and thus does not guarantee satisfactory video quality in case of high PER. Moreover, the authors admit that the method cannot overcome short-term channel failures caused by the receiver mobility.

Another method, Virtual Bottle Neck (VBN) described in [5] deals with so-called multilayered video streams. This technology allows to transmit a video stream consisting of various layers corresponding to different resolutions. Considering the estimated channel capacity, the authors of [5] prioritize packets corresponding to some layers, which can be transmitted without losses and thus prevent the video quality from degradation.

Both the IFD and VBN methods drop less important packets to avoid reaching the queue size threshold without extra protection of other packets.

In contrary, [6] studies various methods of video packet protection and proposes a cross-layer stream protection, which includes forward error correction (FEC) at the application layer, retry limit adaptation and adaptive packet size at the MAC layer. To find the most efficient values for the proposed method parameters, the authors develop an analytic model. However, the authors estimate the delay by considering only the time when the packet is being transmitted, and omitting the time that the packet spends in the queue. Also they assume that both the channel properties and the stream bitrate remain constant.

All the papers describe above consider a stationary channel and do not take into account transients. The first paper found in literature and rising a problem of video quality degradation caused by periodical network faults is [7]. The paper focuses on the robust video streaming in the presence of WLAN co-channel interference. The authors study dynamic interference scenarios, in which the system transits between reliable and unreliable states, and propose an analytic model to calculate packet loss and outage duration because of RTP buffer overflow. However, the model is developed under rather unrealistic assumptions: (i) losses occur only when RTP buffer overflows, which means at least infinite retry limit at MAC layer, and (ii) data rate is constant, while actually video is VBR traffic.

Inspired by this paper, we study the influence of short-term channel failures on the received video quality and develop a method to improve it.

3 Problem Statement

In the paper, we consider real-time video streaming from an Access Point (AP) to several stations (STAs) in the presence of short-term channel failures. The quality of links between the AP and each STA changes in time. Specifically, we consider the case when the available network capacity is enough to deliver video for all the STAs, except for short time intervals with channel failures. Further, we refer to these intervals as *bad intervals*.

Consider a situation when the AP operates by default, i.e. the FIFO queue management policy and ARQ mechanism with predefined retry limit RL.

When the channel degrades for a STA, the AP cannot deliver packets to this STA. However it is trying. After RL unsuccessful retries, it drops the packet and starts serving the next packet in the queue. If this packet is destined for the same STA, the described actions are repeated. Thus, when the duration of the channel failure is more than the time needed to do all $RL + 1$ transmission attempts, one or several video packets may be lost and the quality of video received at the STA with bad channel degrades.

However, this is only the tip of the iceberg. Apart from the fact that the packets destined for the STA with bad channel are not delivered, the AP spends much time to serve them. Indeed, it does the maximum number of retries, not saying that exponential backoff used in Wi-Fi networks increases the time between attempts. So, other packets in the queue are delayed. As the delivery time is an important QoS index and video packets shall be delivered before the deadline called the delay bound in the IEEE 802.11 standard [8], packets not delivered in time are discarded at the receiver. Thus, the channel failure for a STA may degrade the quality of videos for all other STAs.

The goal of the paper is to propose a method to maximize the number of STAs receiving video streams with satisfactory quality. Having maximized the number of STAs with satisfactory video quality, we should maximize overall quality of all video streams.

Let us state the problem more formally. Let $q_i \in [0, 1]$ be the quality of the video stream received at STA i: $q_i = 1$ means ideal quality and $q_i = 0$ corresponds to completely unacceptable quality. Given threshold q_0, we consider the quality as satisfactory if $q_i \geqslant q_0$. So, the task is to maximize the value of the following function:

$$f = \alpha \sum_i \mathbb{1}\{q_i \geqslant q_0\} + \sum_i q_i,$$

where $\mathbb{1}\{q_i \geqslant q_0\}$ is an indicator equal to 1 if $q_i \geqslant q_0$, and equal to 0, otherwise. The first sum corresponds to the contribution from STAs receiving video with satisfactory quality. Parameter α is chosen large enough (e.g., $\alpha > n$, where n is the total number of STAs) to ensure that satisfactory quality for at least one STA contributes to the value of f more than the second sum.

To determine the video quality metric q_i, let us consider how packet losses influence on video quality. When some video packets are lost, one or several frames may be entirely or partially distorted.

When the body of the frame is damaged, but the header is safe, the frame is displayed with artifacts. All frames which refer to the damaged frame for motion compensation are displayed with artifacts, too.

When the header of a frame is damaged or lost, the frame cannot be displayed. Such a frame loss causes freezing and jerking of the video. If the lost frame is not referred to by other frames for motion compensation, the viewer cannot see video distortion except for freezing. In contrary, if the lost frame is used by other frames, these frames are shown with artifacts.

This brief analysis leads us to the idea of video quality estimation with two-component metric: one component corresponds to the video distortion while another one is used to estimate smoothness of video.

We use Mean Square Error (MSE) as the first component and the number L of lost frames in received video as the second one. The MSE value is calculated as follows.

Consider two video samples: original and received (corrupted) ones. The received sample contains N_c frames, while the original one contains N_0 frames. Strictly speaking, N_0 may be greater than N_c, that is, $L \equiv N_0 - N_c \geqslant 0$, but to compute the MSE value we do not take into account frames from original sample which are not displayed in the received one. We take into account the impact of frame losses by calculation of L value.

The MSE value for two frames, original X and received Y, both of the same resolution $W \times H$, is defined as follows:

$$MSE_n = \frac{1}{WH} \sum_{w=1}^{W} \sum_{h=1}^{H} |I_c^n(w,h) - I_0^n(w,h)|^2,$$

where I_0^n and I_c^n are the n-th frames of the original or received sample, and $I_*^n(w,h)$ is the luminance component of pixel with coordinates (w,h). If frames X and Y are identical, MSE equals 0. The MSE value grows with the amount of distortion.

We define MSE value for a video sample as the sum of MSE for all frames:

$$MSE = \sum_{n=1}^{N_c} MSE_n.$$

So, we can consider the quality of received video stream as satisfactory if both components are less than predefined thresholds: $MSE \leqslant MSE_0$ and $L \leqslant L_0$. Otherwise, the quality is unsatisfactory.

4 Proposed Method

The developed method is based on the following ideas.

1. To save the video quality at the STAs with good channel, the AP shall not reduce the percentage of resources spent for each stream destined for such STAs.
2. To improve the video quality at the STAs with bad channel, the AP shall not drop packets when the retry limit threshold is reached, allowing to transmit the packet until they are successfully transmitted or their delay bound is reached.

To combine these conflicting ideas together, we have switched paradigm from 1-persistent queue used by default to p-persistent queue, which operates as follows.

When the AP needs to select the next packet from the queue to be served, it looks through the queue, starting from the head-of-line packet. A packet is chosen with some probability p_i defined for each STA i.

If the packet is not chosen, the AP tests next packets, skipping those which are destined for already considered STAs. It is possible that the AP reaches the end of the queue with no packet chosen. In this case, the AP starts looking through the queue from the very beginning, forgetting that some STAs have already been considered.

If the packet is chosen, the AP transmits it. If the packet is transmitted successfully, it is removed from the queue. If all $RL + 1$ transmission attempts are unsuccessful, the packet is not removed from the queue. In both cases, the station stops serving this packet. To select the next one, the station looks through the queue again, starting from its head.

Let us show how to obtain p_i.

If the last transmission to STA i has been successful, i.e. the AP managed to transmit it with an admissible number of retries, we assume that the channel is good.

If the last transmission to STA i has been unsuccessful and the AP has stopped serving it after $RL + 1$ transmission attempts, we assume that the channel is bad.

Since the packets destined for STAs with bad channel can not be sent successfully, they do not leave the queue until their deadline expires. So, the head of the queue contains only such packets, and we shall set $p_i < 1$ for STAs with bad channel to prevent packets destined for STAs with good channel from being blocked. For STAs with good channel, we set $p_i = 1$, which means that packets for STA i are served as with the default method.

To find p_i, let us suppose that the queue contains packets destined for k STAs, and the channel for z STAs is bad. We denote the probability that the AP chooses the packet destined for a STA with bad channel as p.

Although the developed method works with video streams of different bit rate, to simplify the explanation we assume that all streams are of the same rate and each STA receives only one stream.

Let T_g and T_b be packet service times when the channel is good and bad, correspondingly. Carefully, T_g is the average time needed to do all necessary tries to transmit a packet via good channel, i.e. the time interval from the beginning of the first attempt till ACK reception. By analogy, T_b is the time needed to make $RL + 1$ tries when the channel is bad.

When the channel is good for all STAs, the transmission of each video stream consumes the same percentage ($\frac{1}{k}$) of available channel resource. When the channel is bad for z STAs, these STAs are served with probability p, while STAs with good channel are served with probability $1 - p$. Taking into consideration that packet service times are different for good and bad channel, we obtain that the percentage of resources consumed by z STAs is

$$\frac{pT_b}{(1-p)T_g + pT_b},$$

which shall not increase with respect to the case when the channel is good for all the STAs:

$$\frac{pT_b}{(1-p)T_g + pT_b} = \frac{z}{k}.$$

Denoting $\eta = \frac{T_g}{T_b}$, we obtain that

$$p = \frac{z\eta}{k + z\eta - z}. \tag{1}$$

Video streams destined for STAs with bad channel should obtain the same percentage of channel resources. Let us consider the first packets of each of these streams, which are in the head of the queue.

The packet of the first stream is chosen with probability p_1. With probability $1 - p_1$, we go to the packet of the second stream and choose this packet with probability p_2. To be fair, it it necessary that

$$p_1 = (1 - p_1)p_2.$$

Similarly, we obtain

$$p_2 = (1 - p_2)p_3,$$

$$\dots$$

$$p_{z-1} = (1 - p_{z-1})p_z.$$

The probability that none of this packets is chosen equals $\prod_{i=1}^{z}(1 - p_i)$. On the other hand, it equals $1 - p$:

$$1 - p = \prod_{i=1}^{z}(1 - p_i).$$

It is easy to show that

$$p_1 = \frac{p}{z},$$

$$p_i = \frac{p_{i-1}}{1 - p_{i-1}}, i = 2 \dots z,$$

yields the solution of the problem.

5 Numerical Results

To validate the proposed method, we use a number of Linux applications, including the ns-3 [9] simulator and Matlab environment [10]. First, using the VLC video player [11] and Wireshark [12] application, we obtain a pcap file containing the initial video stream packets and the time moments when they are sent). The packets stored in the pcap file are transmitted through a simulated network with ns-3. The output is also saved in a pcap file which stores all effects

arising due to transmission over wireless channel, including packet losses and delays. The output pcap file is then streamed by ffplay [13] application to VLC application, which saves the stream as a video file. This video file is analyzed in Matlab environment to obtain the video quality metric components introduced in Section 3.

We carry out an experiment in which we study video streaming from an AP to two STAs. We fix 802.11a PHY rate at 9 Mbps. Streams are identical, but they start at different times, shifted by 5 seconds. The experiment lasts for 30 seconds.

During the experiment, PER equals per_0 for all links, except for the time period when channel failure happens to one of the links. During channel failure, PER for this link equals 1. We choose the value of per_0 as follows. We run simulation for different PER values, the default retry limit value of $RL = 7$ and without any intentional channel failures. The maximum PER value for which video is transmitted without any artifacts is chosen as per_0. Packets are dropped from the queue when reaching the delay bound set equal to 400 ms.

We analyze the MAC queue behavior in different runs of this experiment to obtain reliable estimates. As video traffic is non-stationary, there are splashes in queue size, see Figure 1. We choose to start a bad interval for a link with one of the STAs just before one of the splashes, for example, at $t_{start} = 16.8$ seconds in Figure 1, as it is one the most demonstrable cases for video quality deterioration. We vary channel failure duration τ from 150 ms to 400 ms with the step of 50 ms and run the experiment many times for each τ.

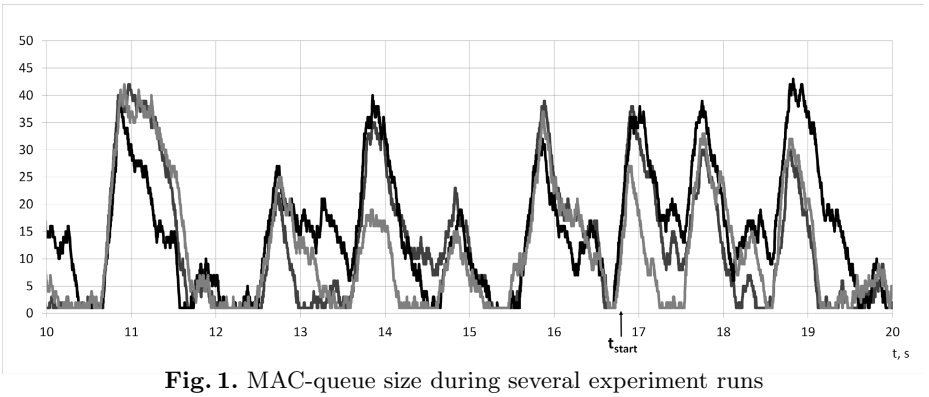


Fig. 1. MAC-queue size during several experiment runs

First, we run experiments for the default queue management policy when $RL = 7$ and for infinite RL. We also run experiments with the proposed p-persistent queue management policy and vary parameter p. The results are shown in Figure 2 for the STA which experiences channel failure during the experiment, and in Figure 3 for the STA without channel failure. We see that if the AP uses the default queue management policy, either with default $RL = 7$ or with infinite RL, it cannot overcome short-channel failure for the first STA with bad channel and also deteriorates the received video quality for the second STA, for which

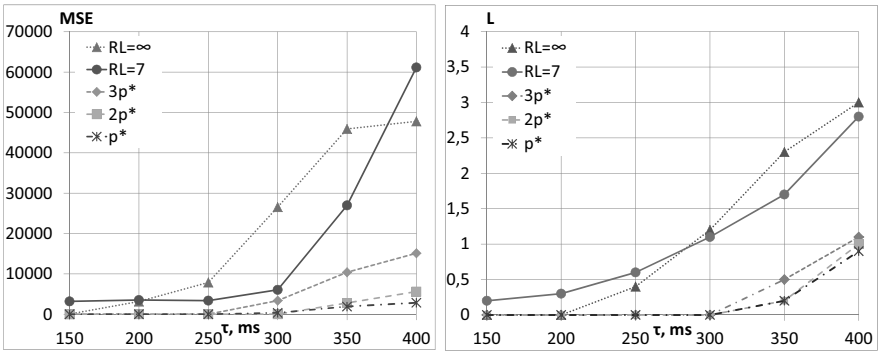


Fig. 2. Metric component values for the first STA

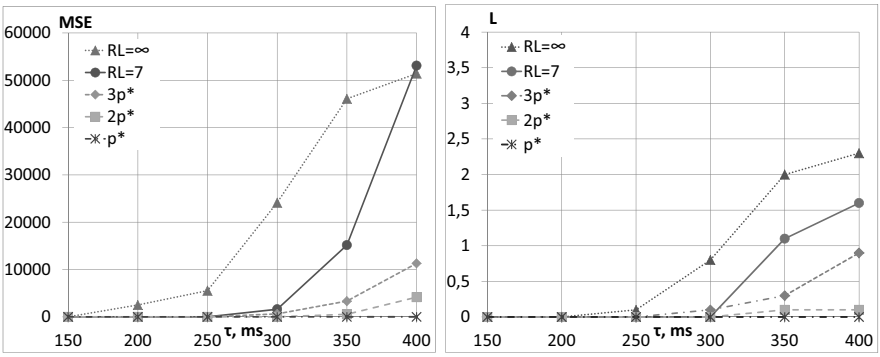


Fig. 3. Metric component values for the second STA

the channel failure does not occur. It happens because of slow packet service for the first STA during the channel failure period (in case of infinite RL, only one packet for the STA is being transmitted all the time). As a result, the queue size grows and packets for both STAs are dropped from the queue because of reaching the delay bound. Both MSE value and the number L of lost frames grow quickly when τ approaches the delay bound equal to 400 ms. If we use our p-persistent queue management policy with the recommended $p = p^*$ value found by (1) in Section 4, the second STA does not experience any video quality degradation (for all $\tau : MSE = 0, L = 0$). As for the first STA, the proposed queue management policy allows to fully overcome channel failure which duration $\tau \leqslant 300$ ms. For $\tau > 300$ ms, our policy also significantly reduces video quality degradation. In Figures 2 and 3, we also show the results for the proposed policy with $p = 2p^*$ and $p = 3p^*$. The usage of this parameter values also shows acceptable results as compared with default method, but the observed video quality is worse than with p^*.

6 Conclusion

Although stations on wireless channel share common medium, channel conditions for the stations may be different at each moment of time as the stations are subject to different errors. In an IEEE 802.11 network, in the case of temporal channel failure for a station, the service for other stations also degrades due to multiple retries and longer backoff times on the failed channel. The core of the problem is in FIFO queue policy: packets are granted the only opportunity to be served and in the order they were queued. In the paper, we propose the p-persistent queue policy when packets are granted multiple opportunities to be served if transmission fails, and the service is granted with probability p, in general case $p < 1$. We also propose a strategy for adaptive tuning of p value and test the strategy for transmission of real-time video streams which are known to comprise non-stationary and heterogeneous traffic. We consider the paper as a very first step and plan to extend the idea of p-persistent queue policy to provide differentiated service for packets of different importance.

Acknowledgement. The reported study was supported by the Ministry of education and science of Russian Federation, (research project No. 8731).

References

1. Bell Labs Video Traffic Study "Video Shakes Up IP Edge", http://www.alcatel-lucent.com/wps/DocumentStreamerServlet?LMSG_CABINET=Docs_and_Resource_Ctr&LMSG_CONTENT_FILE=White_Papers/Video_Shakes_Up_IP_Edge_EN_Whitepaper.pdf
2. Li, Q., van der Schaar, M.: Providing adaptive qos to layered video over wireless local area networks through real-time retry limit adaptation. IEEE Transactions on Multimedia 6(2), 278–290 (2004), doi:10.1109/TMM.2003.822792
3. Kozlov, S., van der Stok, P., Lukkien, J.: Adaptive scheduling of mpeg video frames during real-time wireless video streaming. In: Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, WoWMoM 2005, pp. 460–462 (2005), doi:10.1109/WOWMOM.2005.19
4. Burza, M., Kang, J., Stok, P.: Adaptive streaming of mpeg-based audio/video content over wireless networks. Journal of Multimedia 2(2), http://ojs.academypublisher.com/index.php/jmm/article/view/02021727
5. Bianchi, G., Detti, A., Loreti, P., Pisa, C., Proto, F., Kellerer, W., Thakolsri, S., Widmer, J.: Application-aware h.264 scalable video coding delivery over wireless lan: Experimental assessment. In: Second International Workshop on Cross Layer Design, IWCLD 2009, pp. 1–6 (2009), doi:10.1109/IWCLD.2009.5156512
6. van der Schaar, M., Krishnamachari, S., Choi, S., Xu, X.: Adaptive cross-layer protection strategies for robust scalable video transmission over 802.11 wlans. IEEE Journal on Selected Areas in Communications 21(10), 1752–1763 (2003), doi:10.1109/JSAC.2003.815231
7. Kalogridis, G., Haines, R.J.: A transient reliability model of rtp video streaming over wlan. In: Proceedings of the 7th ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks, PM2HW2N 2012, pp. 149–158. ACM, New York (2012), http://doi.acm.org/10.1145/2387191.2387212, doi:10.1145/2387191.2387212

8. 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 (2012)
9. The ns-3 network simulator, `http://www.nsnam.org/`
10. Matlab simulation environment, `http://www.mathworks.com/`
11. VLC media player, `http://www.videolan.org/vlc/`
12. Wireshark, network protocol analyzer, `http://www.wireshark.org/`
13. FFplay media player, `http://ffmpeg.org/ffplay.html`

Supporting a Pseudo-TDMA Access Scheme in Mesh Wireless Networks

Ilenia Tinnirello and Pierluigi Gallo

DEIM, Universitá di Palermo, Italy
{ilenia.tinnirello,pierluigi.gallo}@unipa.it

Abstract. Wireless mesh networks appear a promising solution for providing ubiquitous low-cost wireless access, but cannot rely on simple CSMA access protocols because of the critical inefficiencies that arise in topologies with hidden nodes. To overcome these limitations, some important protocol extensions based on synchronization and reservation mechanisms have been ratified.

In this paper we show that an alternative approach to the standardization of new features and signaling messages for mesh networks can be the utilization of programmable nodes able to execute different MAC protocols programmed on the fly. Signaling messages are used only for disseminating the new protocol among the nodes. The scheme, that we call pseudo-TDMA, can be optimized as a function of the node density in the network. Apart from the numerical evaluations, we also run some experiments by exploiting our prototype of wireless programmable node called Wireless MAC Processor.

Keywords: wireless mesh networks, synchronization, pseudo-TDMA, random access.

1 Introduction

In recent years, Wireless Mesh Networks (WMNs) based on the IEEE 802.11 technology have gained enormous popularity due to the possibility to provide ubiquitous wireless access to end users with reduced infrastructure costs. However, the original 802.11 standard lacks of several functionalities for effectively managing multi-hop ad-hoc networks, while the legacy DCF channel access protocol has shown significant shortcomings in these network topologies where severe collision rates may arise because of hidden nodes [1].

In order to improve the network transport efficiency and guarantee self-organization, self-configuration, easy installation and maintenance of mesh networks, after an initial proliferation of proprietary incompatible solutions and research proposals, the 802.11s task group has worked on the standardization of new network management functionalities and channel access optimizations. For example, a key component of these enhancements, called MCCA (Multi-user Controlled Channel Access), enables nodes to reserve channel access intervals in advance for avoiding conflicts with contending nodes within a two hops distance.

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 80–92, 2013.

This new feature requires a synchronization mechanism between neighbor nodes and new signaling messages for the set up of the channel reservations.

Obviously, despite the efforts of the standardization group, several brilliant solutions proposed by the research community have not been included in the new ratified functionalities, especially because some solutions are tailored to work in niche network scenarios. However, because of the heterogeneity of mesh network deployments (in terms of scale, node mobility, traffic types, distance to the gateways, etc.) and available hardware (single or multiple radio interfaces, clock stability, directional or omnidirectional antennas, etc.), these networks represent an interesting case in which the traditional concept of *one-for-all* networking solution exhibits clear limitations and new networking paradigms based on node programmability should be considered.

In this paper, after a brief review of a programmable node architecture, called Wireless MAC Processor (WMP), recently proposed for building customized MAC protocols, we focus on the analysis of MAC protocol extensions devised to work in unsynchronized multi-hop networks without signaling overheads. Thanks to the availability of a WMP prototype, we experimentally validate our approach in a simple multi-hop network topology supporting network-level reprogramming. Simulation results have been considered for evaluating the protocol effectiveness in more complex scenarios.

2 Related Work

2.1 The Wireless MAC Processor

The Wireless MAC Processor architecture allows to abstract the heterogeneous hardware capability of the nodes into a set of actions that can be performed on the hardware (starting a frame transmission, detecting the medium activity, freezing/activating a timer, etc.), a set of events triggered by the hardware, and a set of conditions that can be verified on the state of the hardware internal components. The set of hardware actions, events and conditions represent the node API that cannot be modified by the user.

Generic MAC protocols are executed by the MAC Engine, that is an executor of high-level state machines composed on the basis of the hardware abstractions. Indeed, the definition of the medium access control logic in terms of extended finite state machine (XFSM) permits to conveniently control the hardware. In [2] it is shown that completely different medium access operations (including a TDMA, CSMA, multi-channel schemes, and so on) can be defined by exploiting this simple programming model and an API of about ten actions, ten events and ten conditions.

A MAC program is coded into a table of transitions between logical protocol states that is loaded in a memory space on the hardware. Starting from an initial (default) state, the MAC engine fetches the table entry corresponding to the state, and loops until a triggering event associated to that state occurs. It then evaluates the associated conditions on the configuration registers,

and if this is the case, it triggers the associated action and register status updates (if any), executes the state transition, and fetches the new table entry for such destination state.

Since a MAC program is basically a list of labels specifying the events, actions and conditions associated to each state transition, by defining a common set of labels for the API (i.e. a machine language), the MAC program can be transported over data frames from one node to another. In [3] it has been shown that a basic version of DCF can be coded into 500 bytes only. By adding a simple header for controlling the loading and activation of the new state machine on the card, code mobility can be easily supported [3] in the so called MAClets (in analogy to the JAVA applets).

The MAC engine does not need to know to which MAC program a new fetched state belongs, so that a code switching is achieved by moving to a state in a different transition table and by updating the platform configuration registers (e.g. the operating channel, the transmission power, etc.). The definition of code switching transitions are logically independent of the MAC program definition. Therefore, rather than adding them to the MAC program, the architecture allows to program the switching transitions into a second-level state machine (meta state machine), whose states represent the MAC program under execution.

2.2 MCCA Access Mechanism

Mesh networks are characterized by local views of the channel sensed by each node, which have a strong impact on the performance of CSMA protocols both in terms of throughput degradation and in terms of fairness. Assuming that transmission and carrier sense ranges coincide, a transmitting station forces its 1-hop neighbors to be in a frozen state, which in turns will increase the channel access probability of the 2-hop neighbors (not hearing the transmission). As widely documented in literature, the overlapping of frame transmissions originated by stations at a 2-hop distance can lead to severe collisions or to the starvation of some traffic flows with a consequent advantage for some others [4].

To mitigate this problem, the recently introduced Mesh Coordinated Channel Access (MCCA) [5] tries to pre-allocate channel holding times to different groups of nodes for avoiding simultaneous transmissions by 2-hop nodes. MCCA reservations are then propagated to 2-hop neighbors for preventing conflicting allocations by hidden nodes. MCCA access rules provides transmission grants in terms of transmission opportunities (i.e. channel holding times), called MCCAOPs, that are allocated within a multiple of beacon intervals (i.e. a DTIM interval). Each allocation is expressed in terms of: i) *MCCA periodicity*, that is the number of transmission opportunities provided to a given station (with equally space temporal intervals from one opportunity to the next one), ii) *offset* from the starting of the DTIM interval, and iii) *duration* specifying the channel holding time of each transmission opportunity. MCCA allocations are advertised by the transmitter and receiver nodes to their neighbors, which in turn re-broadcast the advertisements to reach the nodes at a distance of 2-hops.

Although the scheme is effective in mitigating hidden node problems and supporting bandwidth reservations in flat topologies, it depends on node synchronization and extra signaling. Commercial wireless cards are equipped with low-quality oscillators, with clock skews ranging from one to one hundred $\mu s/s$. The tradeoff between successful reservations and additional signaling overheads might vary as a function of the node density, source rates and burstiness, and traffic paths. Moreover, as discussed in [6], it still suffers of collisions due to acknowledgement transmissions.

3 Pseudo-TDMA for Wireless Mesh Networks

A critical aspect of MAC protocol schemes for mesh networks is providing synchronization between different nodes. Indeed, network-wide synchronization can be obtained by means of out-of-band signaling employing GPS devices (but this solutions has additional costs and does not work in indoor environments) or by the native 802.11 time synchronization function (TSF) which has been shown to have some scalability problems [7].

To avoid to rely on a global synchronization function, alternative coordination mechanisms among the nodes can employ traffic rate limitations, multi-channel solutions [8], token-based channel grants [9], traffic aggregation [10], and network coding [11]. Although these solutions have not been included in 802.11s, they can be easily supported by nodes based on the WMP architecture. To prove such a feasibility and quantify the achievable performance benefits, we focus on a scheme, that we call pseudo-TDMA, that limits the channel access rate at each node. The scheme supports the allocation of different channel holding times to groups of non interfering stations without explicit negotiation among adjacent nodes.

3.1 Pseudo-TDMA Access Mechanism

Pseudo-TDMA transmissions are performed after a successful random access phase. When a node successfully transmits a packet and receives the relevant acknowledgement (i.e. the intended receiver have not experienced any interfering signal during the whole frame reception process), it assumes to periodically perform subsequent channel accesses for the same traffic flow at regular time intervals. In other words, the first contention acts as a reservation phase after which the channel holding time of the first packet transmission (i.e. the *pseudo-slot*) is considered allocated for the next allocation intervals (i.e. the *pseudo-frames*). In case of successful pseudo-slot allocation, if no other station is trying to reserve a channel holding time, collisions cannot occur. Conversely, when new reservations are performed or reservations are still in progress, pseudo-slots can be affected by collisions. Carrier sense is still used before accessing each pseudo-slot, but no additional backoff is required. When the medium is sensed busy in an allocated pseudo-slot, the transmission is not performed. In such a case or in case of

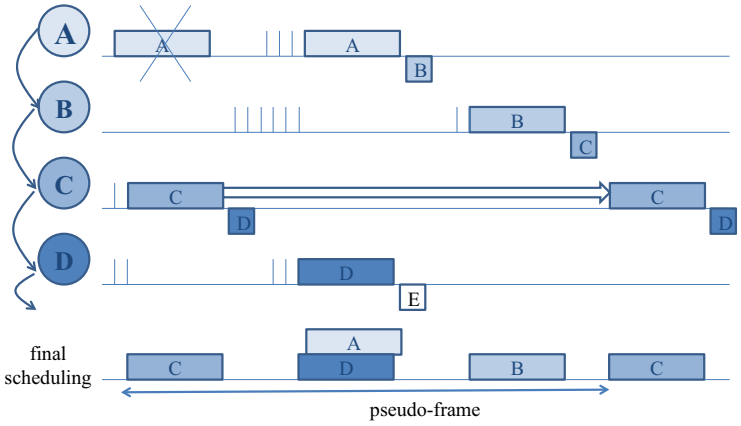


Fig. 1. An example of Pseudo-TDMA access operations in a chain of nodes

collision, the station keeps the slot or tries a new reservation according to a random probability. Note that pseudo-slot allocations are performed on a per-link basis: a given node transporting two different traffic flows towards two different receivers has to allocate two different pseudo-slots, because the interfering conditions depend on the specific receiver location.

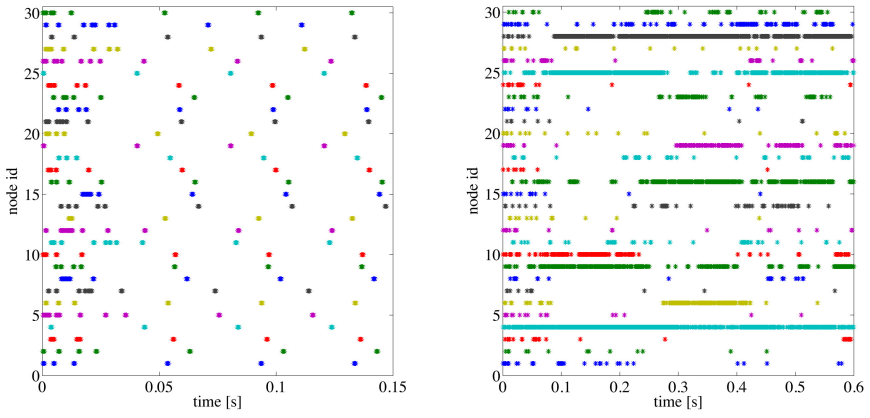
Figure 1 shows an illustrative example of pseudo-TDMA access operations in a network topology given by a chain of nodes. Nodes are labeled from A (first node) to E (last receiver node, not indicated in the figure), while traffic flows are set unidirectionally from two adjacent nodes (from A to B, from B to C, and so on). Each node hears only its neighbors (e.g. B hears only A and C). During the initial random access phase, nodes A and C transmit simultaneously under the assumption that transmission range and carrier sense range coincide. Since node D does not experience any interfering signals, it acknowledges node C transmission. Therefore, node C suspends the random access as indicated by the white arrow, waiting for the next pseudo-slot. As soon as the other nodes perform their first successful transmission, the channel access sequence is repeated periodically according to the final schedule of channel access grants.

As evident from the figure, the scheme basically works by trying to randomly find a successful scheduling of transmissions that can be performed sequentially or in parallel by multiple nodes (C, D+A, B) and repeat such a scheduling over time. Whenever the time interval between successive pseudo-slot allocations (i.e. the pseudo-frame) is large enough to accommodate all the interfering transmissions that cannot be performed simultaneously, after an initial random phase, the frame transmissions occur at regular time intervals without collisions. Note that simultaneous transmissions do not need to be perfectly synchronized (e.g. A and D transmissions), since pseudo-slots are automatically spaced of the time interval required for avoiding interference with 1-hop and 2-hops neighbors.

3.2 Numerical Results

Before testing the pseudo-TDMA performance over the WMP, we performed some simulations for evaluating the scheme performance and scalability in general topologies with a large number of network nodes. Simulations have been performed in MATLAB, where we implemented the generation of random topologies, the setting of random traffic flows, and the tracking of channel access operations under standard DCF or pseudo-TDMA access rules. We quantified the per-node throughput results, as well as the channel access fairness, for different pseudo-frame intervals, in order to study the scheme effectiveness in improving starvation while keeping a good channel utilization. Indeed, for very large pseudo-frame intervals, we can easily imagine that all the nodes can successfully access the channel, but with a very poor throughput performance. Conversely, when the pseudo-frame interval is too short, it might happen that the random access phase is never concluded and some stations are prevented from accessing the channel.

The transient behavior of the scheme is shown in figure 2 for a random topology of 30 nodes uniformly distributed over an area of 300 m $\times$ 300 m with a transmission range (equal to the carrier sense range) set to 100 m. Each node has a greedy traffic source towards a given neighbor node (randomly extracted among the available ones). The figure labels each node with a different identifier and plots the relevant transmission intervals with a different color over a simulation time of 0.15s. For comparison, the figure also shows the transmission intervals of the nodes under legacy DCF.



(a) (b)

Fig. 2. Channel access intervals under pseudo-TDMA (a) and DCF random access (b)

Simulations have been carried out by considering an 802.11b PHY, with a data rate set to 11 Mbps. Under these settings, for a packet payload of 1000 bytes, the time required for transmitting a frame and the acknowledgement is about 1.3 ms. For a fully connected topology with 30 nodes, a perfect TDMA access would require a frame of about 39 ms. Obviously, such a time can be lower when parallel transmissions are possible, provided that a perfect synchronization is available at all the nodes and a central scheduler (aware of interference conflicts) notifies the slot allocations to each node. In figure 2, the pseudo-TDMA scheme has been run with a pseudo-frame of 50 ms. Although such a time is higher than the time required in the fully connected topology, it is large enough to find a final successful scheduling by the end of the first pseudo-frame. All the stations succeed in accessing the channel, without the starvation effects evident for legacy DCF (e.g. most of the time the channel is used only by stations 28, 25, 16, 10, 9 and 4).

Table 1. Simulation parameters

Common	
# nodes	30
topology	uniformly distributed nodes over a square
traffic model	saturated sources
dst selection	randomly choosen among neighbors
pkt duration	1 ms

Pseudo-TDMA		DCF random access	
# of pseudo-TDMA slots per frame	30	CW_{min}	7
		CW_{max}	1023

We run experiments for 20 different topologies with similar characteristics (30 nodes, deployment area of 300 m $\times$ 300 m, transmission range set to 100 m) and averaged the aggregated and per-node throughput performance. Figure 3-(a) and (b) plots the total number of successfully transmitted packets and collided packets in the network for pseudo-TDMA with different pseudo-frame intervals (a) and DCF with different contention windows (b) in a simulation run of 1 s. For pseudo-TDMA the figure adopts a logarithmic scale, while the pseudo-frame interval is measured as a multiple of the channel holding time required for transmitting and acknowledging a packet (i.e. multiple of pseudo-slots). For a pseudo-frame higher than 30 pseudo-slots the fairness index is about one, while the throughput obviously degrades because of longer intervals in which the channel remains idle. Note also that in our network topology nodes have a limited distance (in hops) and therefore parallel (non-interfering) transmissions are not likely to occur.

If we compare the pseudo-TDMA performance with legacy DCF we can immediately observe that even with short pseudo-frame intervals pseudo-TDMA is more fair. Conversely, DCF can achieve higher aggregated throughput, but this throughput is shared by a few nodes with several flows suffering starvation.

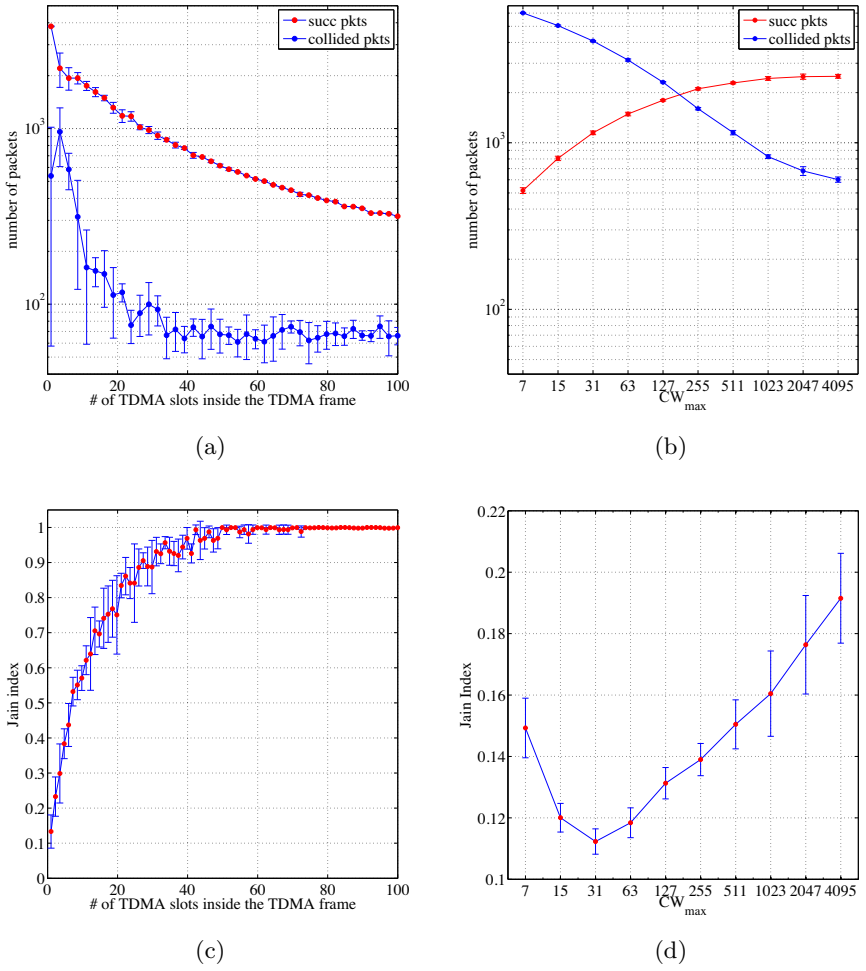
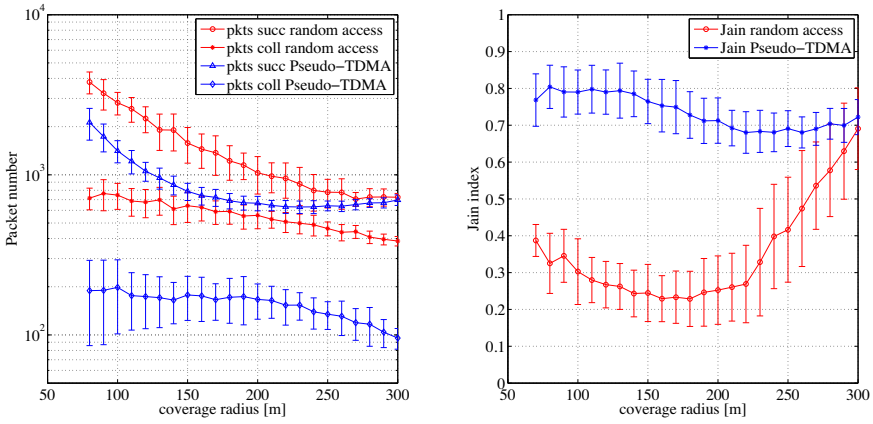


Fig. 3. Comparison between pseudo-TDMA and DCF: successful and collided packets (a), (b) and fairness (Jain) index (c), (d).

Even reducing the DCF access rate (using higher collision windows) the fairness performance does not improve significantly.

Finally, figure 4 compares successful and collided packets in (a), and fairness index in (b) for different graph depths obtained varying the coverage radius of each node range from 80 m to 300 m. In figure, the number of pseudo-TDMA slots in a pseudo-TDMA frame is computed as the next integer after the average degree of the square of the connectivity graph (G^2). The average degree of G^2 can be proficiently used to dimension the pseudo-TDMA frame because nodes have enough independent resources in terms of pseudo-TDMA slots to avoid



(a) (b)

Fig. 4. Performance comparison between pseudo-TDMA (number of slots is the next integer after the average degree of G^2) and random access: pkts successful or collided (a), Jain index (b).

collisions with neighbors up to two hops. Under such conditions, figure 4-(a) shows that pseudo-TDMA has an aggregate throughput comparable to DCF when the graph is highly connected. When the radius coverage is smaller, the aggregate number of successfully transmitted packets is lower than the DCF one but the Jain index is notably higher. Pseudo-TDMA always guarantees the best fairness and the lowest collision rates, which also implies energy savings and reduced airtime waste.

4 Experimental Results

In order to experimentally validate the proposed scheme in network topologies non fully connected, we designed a MAC protocol state machine on the basis of the WMP API. The state machine has been obtained by slightly modifying the one presented in [2], which in turns integrates a few modifications to the basic DCF transmission state machine. Figure 5 shows the resulting state machine. After the first ACK reception, the transition to the TX state can be performed from the WAIT PSEUDO FRAME state at the expiration of the frame interval timer. Since the medium state is verified before performing such a transition, in case of medium busy the machine remains to the WAIT PSEUDO FRAME state or switch to a new random access phase (BACKOFF state) with probability P_{new}. In case of empty queue, the machine comes back to the IDLE state. The diagram reports in blue the modifications compared to the DCF.

In [2] the accuracy of the pseudo-frame regular scheduling has been verified by analyzing the channel activity traces acquired by a USRP. We verified that also

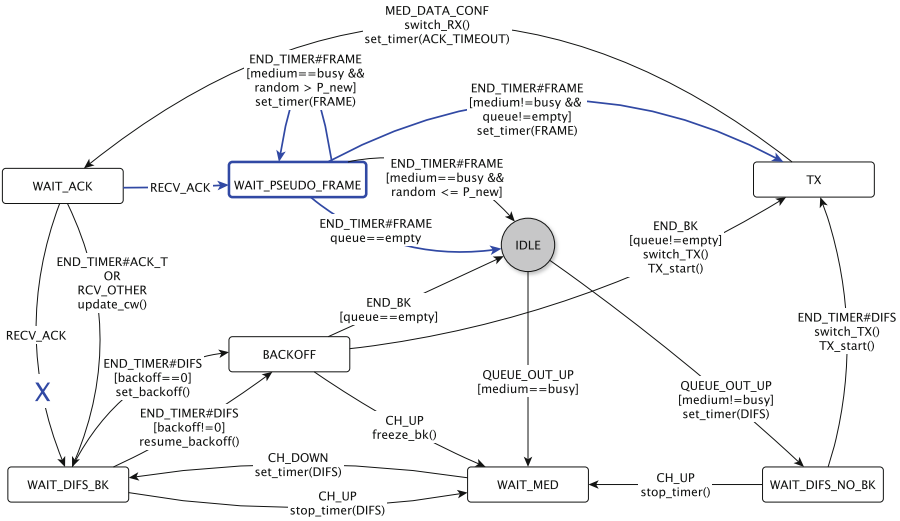


Fig. 5. Pseudo-TDMA implementation in terms of adjustments on DCF state machine.

with this implementation the medium access times are scheduled with a precision of the order of micro-seconds (not achievable with driver level hackings).

To run our experiments we considered a simple network topology with a chain of three nodes (labeled as STA-A, STA-B and STA-C). Node B can hear both the other stations, while nodes A and C are hidden to each other. Moreover, the propagation conditions of stations A and C towards node B are not symmetrical, and in particular node C suffers of higher attenuation levels and channel variability (likely due to the lack of line of sight propagation). Indeed, the three nodes have been placed in three different rooms; the distance between STA-A and STA-B is about 16 m (through two walls), while the distance between STA-B and STA-C is about 20 m (through four walls). We run two experiments under legacy DCF and under pseudo-TDMA, by considering the throughput results of node A and node C towards node B (that acts as a common receiver). For each experiment, we collected results when only node A or C are active and when the two nodes are simultaneously active. In figure 6 we plot the throughput results (DCF in the three top figures and pseudo-TDMA in the three bottom figures). In case of legacy DCF, when the two stations are simultaneously active the throughput results of the two stations are strongly unbalanced. Indeed, node A and node C transmissions often overlap (being the two nodes unable to sense each other), but since node A transmissions are received by node B with a much higher power than node B transmissions, collisions result in an exact demodulation of node A packets. This phenomenon, that is known in literature as *capture*, is clearly evident from figure 6-(c), where node A throughput is slightly lower than the throughput obtained when node C is off. Note that in case of visible contending nodes this throughput would have been about one half of the

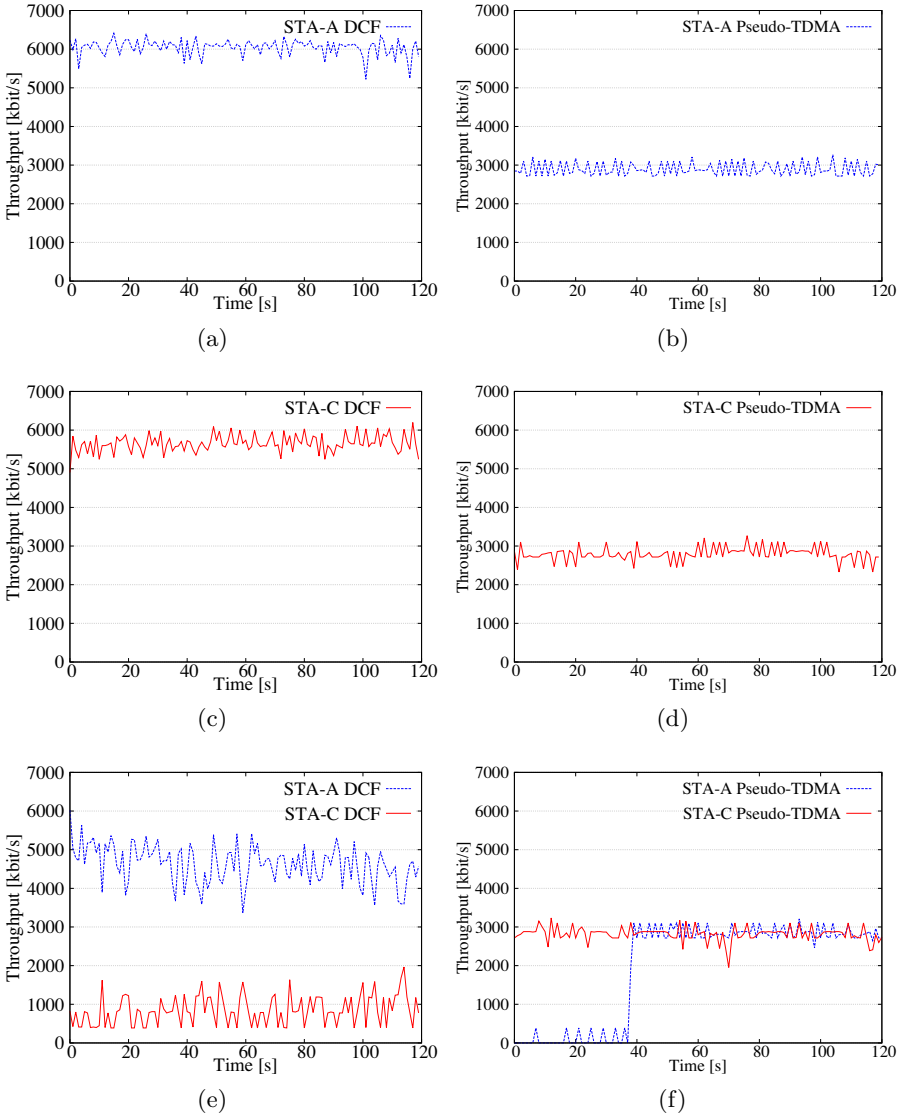


Fig. 6. Throughput performance of a chain of nodes when the node at the edge of the chain transmit to the middle node under DCF (a),(b),(c) and pseudo-TDMA (d),(e),(f).

throughput obtained with a single contending node (namely, about 6 Mbps for a packet size of 1500 bytes and a data rate of 11Mbps). In case of pseudo-TDMA with a pseudo-frame set to 4ms (able to accommodate the transmission of two packets and acknowledgements in every pseudo-frame interval), after a transient phase, the two stations equally share the available bandwidth when they are

simultaneously active, with a throughput equal to 12000bit/4ms=3Mbps. Note that, as described in [3], the switching from DCF to pseudo-TDMA could be automatically programmed into a meta-state machine in case of high collision rates.

5 Conclusions

In this paper we propose a simple extension of legacy DCF devised to work in mesh networks without requiring node synchronization and reservation messages. The basic idea of the protocol is combining random access and regular scheduling of packet transmissions, in order to repeat the sequence of channel accesses which result successful. Provided that the scheduling interval is large enough, the scheme is able to allocate one packet transmission to each contending node without signaling messages, while preventing flow starvation (as verified via simulation). Despite its simplicity the scheme cannot be supported in legacy 802.11 cards. However, thanks to the availability of a card able to execute generic MAC protocols programmed in terms of state machines (the so called Wireless MAC Processor), we implemented very easily the scheme and run some experiments in a simple network topology. More interesting, whenever the mesh nodes implement the WMP architecture, such a scheme can be dynamically programmed (or reconfigured, e.g. changing the pseudo-frame interval) by simply flooding a data packet transporting the protocol state machine in the whole network [3]. The experimental validation of our approach is performed in a simple multi-hop network topology supporting network-level reprogramming, further experiments will be run in the CREW testbed [17].

Acknowledgments. This work has been supported in part by the EU projects FP7-257263 (FLAVIA) and FP7-258301 (CABIN-CREW).

References

1. Xu, S., and Saadawi, T.: Revealing the problems with 802.11 medium access control protocol in multi-hop wireless ad hoc networks. Computer Networks 38, no. 4, pp. 531-548, 2002.
2. Tinnirello, I., Bianchi, G., Gallo, P., Garlisi, D., Giuliano, F. and Gringoli. F.: Wireless mac processors: Programming mac protocols on commodity hardware. In INFOCOM, 2012 Proceedings IEEE, pp. 1269-1277. IEEE, 2012.
3. Bianchi, G., Gallo, P., Garlisi, D., Giuliano, F., Gringoli, F., and Tinnirello, I.: MAClets: active MAC protocols over hard-coded devices. In Proceedings of the 8th international conference on Emerging networking experiments and technologies, pp. 229-240. ACM, 2012.
4. Lyakhov, A., Pustogarov, I., Safonov, A., and Yakimov, M.: Starvation effect study in IEEE 802.11 mesh networks. In Proc. of Third IEEE International Workshop on Enabling Technologies and Standards for Wireless Mesh Networking (MeshTech'09), Macao SAR, P.R. China, 2009

5. IEEE Standard for Information technology *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. 6 February 2012
6. Krasilov, A., Lyakhov, A., and Safonov., A.: Interference, even with MCCA channel access method in IEEE 802.11s mesh networks. In Mobile Adhoc and Sensor Systems (MASS), 2011, pp. 752-757. IEEE, 2011.
7. Huang, L., and Lai, T.H.: On the scalability of IEEE 802.11 ad hoc networks. In Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, pp. 173-182. ACM, 2002.
8. Shi, J., Salonidis, T., and Knightly, E.W.: Starvation mitigation through multi-channel coordination in CSMA multi-hop wireless networks. In Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing, pp. 214-225. ACM, 2006.
9. Tinnirello, I., Scalia, L., and Campoccia, F.: Improving IEEE 802.11 performance in chain topologies through distributed polling and network coding. In IEEE ICC'09, pp. 1-6. IEEE, 2009.
10. Tao, Z., Teo, K.H., and Zhang, J.: Aggregation and concatenation in IEEE 802. 16j mobile multihop relay (MMR) networks. In Mobile WiMAX Symposium, 2007. IEEE, pp. 85-90. IEEE, 2007.
11. Katti, S., Rahul, H., Hu, W., Katabi, D., Mdard, M. and Crowcroft, J.: XORs in the air: practical wireless network coding. In ACM SIGCOMM Computer Communication Review, vol. 36, no. 4, pp. 243-254. ACM, 2006.
12. Xu, S., and Saadawi, T.: Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?. Communications Magazine, IEEE 39, no. 6, pp 130-137, 2001.
13. Akyildiz, I. F., Wang, W., and Wang, W.: Wireless mesh networks: a survey. Computer networks 47, no. 4 (2005): 445-487.
14. He, Y., Yuan, R., Sun, J., and Gong, W.: Semi-Random Backoff: Towards resource reservation for channel access in wireless LANs. In 17th ICNP 2009 pp. 21-30. IEEE, 2009.
15. Koutsonikolas, D., Salonidis, T., Lundgren, H., LeGuyadec, P., Hu, Y.C., and Sheriff, I.: TDM MAC protocol design and implementation for wireless mesh networks. In Proceedings of the 2008 ACM CoNEXT Conference, p. 28. ACM, 2008.
16. Djukic, P., and Mohapatra, P.: Soft-TDMAC: A software TDMA-based MAC over commodity 802.11 hardware. In INFOCOM 2009, pp. 1836-1844. IEEE, 2009.
17. CREW, Cognitive Radio Experimentation World, http://www.crew-project.eu/

Dynamic Resource Allocation for MCCA-Based Streaming in Wi-Fi Mesh Networks

Evgeny Khorov, Artem Krasilov, Andrey Lyakhov, and Dmitry Ostrovsky

Institute for Information Transmission Problems of Russian Academy of Sciences (IITP RAS), Bolshoy Karetny per. 19, Moscow, 127994, Russia
{khorov,krasilov,lyakhov,ostrovsky}@iitp.ru

Abstract. The IEEE 802.11s standard defines a novel deterministic channel access method called MCCA. In preliminarily reserved time intervals, MCCA allows packet transmission from the owner of the reservation to the intended receiver, while their neighbors are forbidden to access the channel. Such a protection decreases the interference from hidden stations, allowing to improve reliability of transmission between two stations, which makes MCCA a promising method to transmit real-time multimedia streams with parameterized QoS support. However, both random noise and interference from the stations out of one-hop neighborhood may cause transmission errors, increasing packet loss ratio. In this paper, we consider various mechanisms for protecting reservations from interference and propose a method to find the amount of reserved channel resources and the type of protection needed to meet QoS requirements while transmitting the data with MCCA in case of varying channel conditions.

Keywords: Wi-Fi Mesh, MCCA, deterministic channel access method, paremetrized QoS.

1 Introduction

Wireless networking technologies are rapidly evolving. Apart from the growth of such apparent indices as capacity or throughput, wireless networks become more intelligent: self-organization, robustness, and parameterized QoS support for real-time multimedia traffic are today's issues. Another evidence of this evolution is the expansion of wireless technologies to those scenarios which the technologies were not initially intended for, e.g. the increase of Wi-Fi network coverage area by means of multihop operation or the usage of LTE femtocells as local area networks.

Understanding the trends in the wireless networking market, IEEE 802.11 standards committee paid attention to self-organizing multihop networks and developed the IEEE 802.11s standard [1] defining Wi-Fi Mesh which seems to be a promising technology to expand the coverage area of wireless local area networks (WLANs) and to increase their robustness. The standard is more than just a routing framework. It defines a novel deterministic channel access method

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 93–111, 2013.

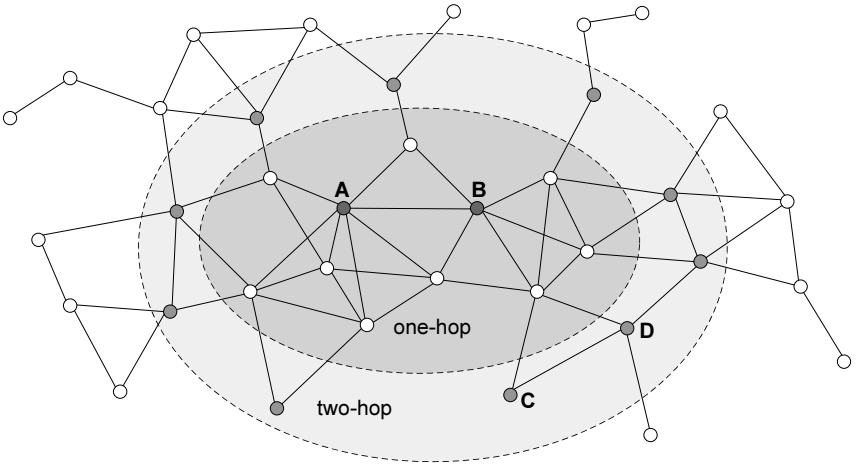


Fig. 1. One-hop and two-hop neighborhood of two stations A and B

called Mesh coordination function Controlled Channel Access (MCCA), which can be used in addition to the random access method traditional to Wi-Fi. The reason to introduce a new channel access method is clear: the performance of random access in multihop networks significantly degrades with traffic growth because of the hidden station problem [2]. So, random access is incapable of providing high throughput and low packet loss ratio (PLR), which is strictly necessary to meet QoS requirements while transmitting real-time multimedia streams.

The key idea of MCCA is preliminary reservation of time intervals during which a station transmits packets to another one, while neighbors of the stations are forbidden to transmit packets. Defining a method but not the rules of its usage, the standard opens the door for many papers studying its efficiency and proposing algorithms for reservation allocation. In particular, recent studies [3,4] show that in some scenarios, MCCA does significantly reduce transmission errors, but does not entirely eliminate them. Transmission attempts may be unsuccessful due to random noise or interference caused by stations out of one-hop neighborhood. To reduce PLR, [4] proposes to use two-hop protection, i.e. to restrict transmission in two-hop neighborhood, see Fig. 1. Naturally, the usage of two-hop protection limits spatial reuse and thus decreases network capacity; so it is necessary to find a trade off between the low PLR and the waste of channel resources (more exactly, channel time). In other words, the type of reservation protection should be dynamically chosen, according to the current channel conditions. Another way to reduce PLR is to reserve more resources for additional transmission attempts.

This paper addresses the issue of determining the amount of channel resources and the type of protection needed to provide QoS support for a multimedia stream in case of dynamic channel conditions.

To simplify our analysis we assume that:

1. The routing protocol, e.g. developed in [5], has found a stable path between the source and destination stations.
2. The end-to-end QoS requirements (PLR and delay) are divided among all hops in the path, e.g. according to the strategy presented in [6].

So, we need to develop a method to obtain the amount of reserved channel resources with various protection types for *the single hop case*. This is rather a challenging task, since no solutions have been developed yet even for the only protection type.

The rest of the paper is organized as follows. In Section 2, we study the possibility of providing parameterized QoS support with deterministic channel access methods employed by various technologies, focusing on the MCCA open issues, in particular, on the problem of dynamic resource allocation for MCCA-based streaming. We develop our resource allocation method in Section 3. In Section 4, we evaluate its efficiency in various scenarios. Section 5 concludes the paper.

2 The Problem of Dynamic Resource Allocation for MCCA-Based Streaming

2.1 Parameterized QoS Support via Deterministic Access: State-of-the-Art

Parameterized QoS support in wireless networks is a challenging and still not completely solved task. Most of the related solutions proposed in recent years are based on the usage of deterministic channel access methods which allow to reserve exclusive channel resources needed for transmission of streams with specific QoS requirements. To allocate resources for deterministic access, various technologies employ different resource reservation schemes, which can be classified as *centralized* and *distributed* [7]. With centralized schemes, resource scheduling is entirely conducted by coordinators, e.g. Access Points in infrastructure IEEE 802.11 WLANs or Base Stations in IEEE 802.16 and 3GPP LTE networks. In contrast, with distributed schemes, there is no dedicated coordinator and resources are allocated by means of a special negotiation procedure which involves transmission of control messages between the intended transmitter and receiver and also their neighbors. For instance, distributed reservation schemes are implemented in IEEE 802.11s MCCA and ECMA 368 Distributed Reservation Protocol (DRP) [8].

To date, many research works have been focused on providing parameterized QoS support via centralized schemes. For example, dozens of algorithms have been developed to allocate channel resources for Constant Bit Rate (CBR), e.g. reference solution in [9], or Variable Bit Rate (VBR, e.g. video) [10, 11] streams with IEEE 802.11 HCCA[1]. Unfortunately, these algorithms cannot be directly

[1] Hybrid coordination function Controlled Channel Access (HCCA) implements a centralized resource reservation scheme in infrastructure IEEE 802.11 WLANs.

used in mesh networking, since they require reallocation of reserved resources on the fly, which is provided in centralized schemes by the ability of the coordinator to fully control the wireless channel. On the contrary, with distributed reservation schemes which are usually applied in mesh networks, reallocation of resources implies a long negotiation procedure not taken into account by the mentioned above algorithms.

As for distributed reservation schemes, a number of papers present resource allocation algorithms for ECMA 368 DRP. Based on the prediction of the maximum incoming bit rate [12] or traffic specification provided from the upper layers [13], the algorithms estimate the amount of resources needed to transmit CBR and VBR streams and allocate resources accordingly. However, these algorithms assume that all transmissions in reserved time slots are successful, which is not always valid due to error-prone nature of the wireless channel and interference from other stations [4].

In [14] the authors have presented a mathematical model of MCCA-based streaming in IEEE 802.11s networks. The model allows to estimate the amount of resources needed to be reserved for transmission of a multimedia stream with given QoS requirements, assuming that packet transmissions may fail with a fixed probability due to noise. However, the authors have not provided any guidelines on how to estimate this probability and how to reallocate resources if channel conditions change. In this paper, we address these issues and propose an adaptive method which periodically estimates channel conditions and dynamically allocates resources to meet QoS requirements of a transmitted stream.

2.2 MCCA-Based Streaming

In this section, we consider how to organize transmission of QoS-sensitive stream with MCCA. Although the interested reader can study the standard [1] or at least [4] to get detailed information about MCCA, let us give its brief description to simplify the reading of the paper. As mentioned in Section 1, MCCA is a channel access method which allows stations to reserve time intervals, called MCCAOPs, for data transmissions. Specifically, each MCCAOP reservation (further: reservation) represents a strictly periodic set of time intervals of equal duration, in which the owner of the reservation obtains an exclusive access while neighbors of the intended transmitter and receiver shall abstain from transmission.

To establish a new reservation, a station runs a special negotiation procedure which can be shortly described as follows. By exchanging control messages sent with the random channel access method, the intended transmitter and receiver find an appropriate location for a new reservation so that it does not overlap with existing reservations of their neighboring stations. Once the appropriate location is negotiated, the transmitter and the receiver start advertising the established reservation to their neighboring stations, which then re-broadcast the advertisement to ensure that all stations in the two-hop neighborhood become aware of this reservation. According to the standard, stations advertise periodically all established reservations in beacons. Thus, reservations can not be established or

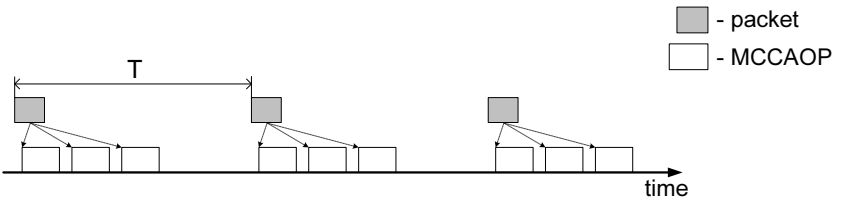


Fig. 2. Proposed resource reservation scheme

released "on the fly", because at least one beacon interval is needed to notify all neighboring stations about the changes.

As reservations provide regular and controlled access to the wireless channel, it is very suitable to use MCCA for transmission of multimedia streams with strict QoS requirements. Typically, QoS requirements are expressed as follows: the delay of each delivered packet of the stream shall not exceed the maximum delay D^{QoS}, and the packet loss ratio (PLR) shall not exceed the threshold PLR^{QoS}.

As an example, let us consider how to organize the transmission of a CBR stream (e.g. voice stream) between two stations with MCCA. Suppose that packets of the stream arrive at the transmitter with a fixed interval T. At first sight, the solution looks very simple: we need to establish a reservation with the interval between consequent MCCAOPs equal to T, and the duration of each MCCAOP sufficient for the transmission of a packet itself and a MAC acknowledgement. However, as already discussed in Section 1, despite all neighbors of the transmitter and the receiver abstaining from accessing the channel during MCCAOP, transmission inside MCCAOP is not always successful, which, in turn, may lead to the violation of the PLR requirement. There are two reasons of transmission failures inside MCCAOP: (i) random noise and (ii) interference caused by stations outside one-hop neighborhood, which is not always negligible as shown in [4].

One of the ways to meet the PLR requirement is to reserve more resources so that failed packets have additional transmission attempts. It is evident that the amount of resources needed to meet the PLR requirement depends on the current channel conditions. As we show in Section 4, the conditions may significantly vary over time because of unstable nature of wireless channel and unpredictable interference from stations outside the one-hop neighborhood of transmitter and receiver. So we need to dynamically reallocate reservations according to the observed channel conditions.

The resources may be reserved as the only series of MCCAOPs, i.e. the only reservation with the period less than T, as it is considered in [14]. However, when we need to reallocate resources we have two options. According to the first one, we release the current reservation and then establish a new one, which leads to a service gap caused by the negotiation procedure. According to the second one, we keep using the old reservation during the negotiation procedure, doubling consumed resources. As both options degrade network performance, we

use another approach shown in Fig 2. According to this approach the station establishes a number of reservations with a period T so that each packet is transmitted a fixed number of times. If we need to change the amount of reserved resources, we just release or establish some reservations, keeping others. Note that with such approach the delay requirement can be considered to be satisfied since all transmission attempts of the same packet can be arranged not farther than D^{QoS} from the packet arrival.

Another way to meet PLR requirement is to increase the reliability of transmission with MCCA by reducing the interference from stations outside the one-hop neighborhood of transmitter and receiver. For this purpose, in [4] it is proposed to forbid all stations in two-hop neighborhood to transmit during MCCAOPs (i.e. to use two-hop protection for reservations instead of one-hop protection specified in the standard). It has been shown that the usage of two-hop protection allows to significantly increase the probability of successful transmission in MCCAOPs, but on the other hand, it decreases network capacity by limiting spatial reuse of the channel. So, the protection type used for a reservation should be chosen according to the current channel conditions. Unfortunately, the standard [1] supports only one-hop protection for all reservations. Thanks to the advances of the ICT FP7 FLAVIA research project [15] which proposes an architectural solution for making the channel access programmable, it is possible to add the required functionality to MAC protocol and easily implement such an adaptive choice [4].

Summing up, in case of the violation of the PLR requirement we have two options which can be applied simultaneously: (i) to establish additional reservations to provide more transmission attempts and (ii) to use two-hop protection instead of one-hop protection for new reservations to improve reliability of transmission in MCCAOPs. The method developed in this paper uses both of these options.

2.3 Problem Statement

Let us define the requirements for our method more strictly.

1. As the use of the resource reservation scheme proposed in Section 2.2 and shown in Fig. 2 leads to meeting the delay requirement, our goal is to develop a method which allows *to meet the PLR requirement*. PLR may change over time, so we consider a series of time intervals of fixed duration and measure PLR for each interval I as the number of lost packets (L_I) divided by the number of sent packets W_I (which is constant for a CBR stream: $W_I \equiv W$) during interval I. At first sight, the PLR requirement may be written as $\frac{L_I}{W} \leq PLR^{QoS}$, but the wireless channel unpredictably changes over time. Adjusted to meet the PLR requirement with current channel conditions, the method may violate it after the conditions change. So, strictly speaking, the PLR requirement should be rewritten as

$$\mathbb{P}\left\{\frac{L_I}{W} \leq PLR^{QoS}\right\} \geq \alpha, \tag{1}$$

where $\mathbb{P}\{A\}$ is the probability of event A and confidence α is close to 1, say 0.95.

2. Providing QoS support, the proposed method shall *consume as few channel resources as possible*. Taking into consideration that channel resource consumption of two-hop protection is higher than that of one-hop protection, we assign various costs to the reservations with different protection types. Let the cost of a reservation with two-hop protection be c_{th} times higher than the cost of a reservation with one-hop protection. For example, c_{th} may equal the ratio of the number of stations in two-hop neighborhood to the number of stations in one-hop neighborhood of transmitter and receiver. We obtain that among all sets of reservations which allow to satisfy (1), the method shall select the set of the lowest possible cost.
3. Finally, we shall take into consideration that the method *shall not fluctuate*, i.e. shall not establish or release reservations too frequently, since every change in the set of reservations increases overhead and may take some time. Indeed, *to establish a new reservation*, the stations run a negotiation procedure and then advertise the established reservation. It takes some time t_{setup} which is about the beacon interval. When a station *releases an existing reservation*, its neighbors become aware of this action after the next advertisement. So, actually, the resources are released only after t_{setup}.

In Section 3, we present the developed method which satisfies the requirements specified above.

3 Proposed Method

3.1 Key Ideas

As described in Section 2.2, the number of reservations (i.e. the number of packet transmission attempts) needed to meet QoS requirements for each particular stream depends on the channel conditions, which may significantly vary over time. So, the key idea of the proposed method consists in the periodical running of the procedure which considers the current set of reservations, estimates channel conditions, and, if needed, adds or removes some reservations, forming the new set. The procedure consists of three steps.

At the first step, we describe the transmission of the stream by a statistical model which allows to find the probability that an arbitrary packet of the stream will be successfully transmitted with set R of reservations as function $P_\theta(R)$ with parameter θ corresponding to current channel conditions. In Section 3.2, we present an example of the statistical model. However, other models may also be used. We use the statistics of all transmission attempts in the current set R^{cur} of reservations for the last h packets to tune the model. In other words, we find $\theta = \hat{\theta}$ which optimally describes channel conditions, using the maximum-likelyhood approach [16].

At the second step, we use the model to find the cheapest set R^* of reservations meeting the PLR requirement. To check if it is met for set R, we find

the probability that for W packets the number of lost packets is less or equal to $\lfloor W \cdot PLR^{QoS} \rfloor$. Assuming that transmissions of various packets are statistically independent, we obtain that this probability equals $F_{W,1-P_{\hat{\theta}}(R)}(\lfloor W \cdot PLR^{QoS} \rfloor)$, where

$$F_{W,1-P_{\hat{\theta}}(R)}(k) = \sum_{i=0}^{k} \frac{W!}{(W-i)!\,i!}(1-P_{\hat{\theta}}(R))^i P_{\hat{\theta}}(R)^{W-i}$$

is the c.d.f. of binomial distribution with W independent experiments. The PLR requirement is satisfied with confidence α if the following condition is true:

$$F_{W,1-P_{\hat{\theta}}(R)}(\lfloor W \cdot PLR^{QoS} \rfloor) \geq \alpha. \tag{2}$$

At the third step, taking recommended set R^* and the statistics for current set R^{cur} into account, we form a new set R^{new} of reservations to be used further. Actually, R^{new} may differ from both R^* and R^{cur}.

The described above procedure is run periodically. The time interval between two executions of the procedure depends on packet interarrival time T and the result of the previous execution. If no reservations were added, next time the procedure is run after τ packets are sent, $\tau \leq h$. If one or several reservations were added, it is necessary to establish them, which takes t_{setup}, and then to obtain statistics for new reservations, which requires h packets to be sent via the new reservations. If T is constant, the procedure is run after $\tau' = \lceil \frac{t_{setup}}{T} \rceil + h$ packets are sent.

To describe the procedure in detail, let us start with the case when all reservations are of the same cost, e.g. only one-hop protection is used for all reservations. For this case, we develop a basic procedure described in Section 3.2 and two enhancements of its third step which improve QoS support and reduce fluctuation (see Sections 3.3 and 3.4, respectively). These enhancements may be used separately or together. We consider the complex case with reservations of various costs in Section 3.5.

3.2 Basic Procedure

At the first step, we tune the statistical model. We develop the model, based on the following ideas:

1. Transmission attempts in various reservations are *statistically independent* and, generally speaking, may have different success probabilities.
2. Success transmission probability for an existing reservation is estimated via the statistics for this reservation.
3. Success transmission probability for a new reservation (i.e. reservation which may be established in addition to existing ones) is approximated as the mean of success probabilities for existing reservations.

Speaking more formally, let p_i be the success transmission probability for existing reservation $i, i = \overline{1,r}$, where $r = |R^{cur}|$ is the size of the current set R^{cur} of reservations. The proposed statistical model is parameterized by vector $\theta =$

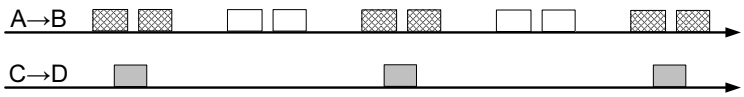


Fig. 3. Correlated transmission errors

$(p_1, .., p_r)$. We fit θ using the maximum-likelyhood approach, which yields the estimate of p_i:

$$\hat{p}_i = \frac{1}{h} \sum_{j=1}^{h} X_{ij}, \quad i = \overline{1, r}, \tag{3}$$

where X is the binary matrix of r rows and h columns, $X_{ij} = 1$ if the transmission of packet j via reservation i was successful.

We estimate $p_{i'}$ for a new reservation $i' > r$ as the mean value of $\hat{p}_i$:

$$\hat{p}_{i'} = \frac{1}{r} \sum_{i=1}^{r} \hat{p}_i. \tag{4}$$

Since transmission attempts in different reservations are assumed to be independent, the probability of successful packet transmission for a given set R of reservations and vector $\hat{\theta} = (\hat{p}_1, .., \hat{p}_r)$ can be expressed as

$$P_{\hat{\theta}}(R) = 1 - \prod_{i \in R} (1 - \hat{p}_i). \tag{5}$$

At the second step, we use the model to obtain the cheapest set R^* of reservations which allows to meet the PLR requirement. For that, if requirement (2) is not met for the current set R^{cur}, we add new reservations one by one until (2) holds. Otherwise, we remove the reservations with the lowest p_i one by one until removing of the reservation breaks (2).

At the third step, we merely follow the statistical model recommendation and choose $R^{new} \leftarrow R^*$. Such a simple step of the basic procedure is enhanced below.

3.3 Enhancement to Improve QoS Support

The statistical model described above operates under the assumption that transmission attempts in various reservations are statistically independent. In fact, this assumption may not hold. Let us consider a simple example shown in Fig. 3. Station A (refer to Fig. 1) transmits a stream to station B via two reservations. The other stream is transmitted via one reservation from C to D, which are both two-hop neighbors of B. Packet interarrival time of stream $C \to D$ is twice as long as that of $A \to B$. In this case, transmission attempts for odd packets of stream $A \to B$ may fail because their MCCAOPs (crossed in Fig. 3) overlap with MCCAOPs of stream $C \to D$, while transmission attempts of even packets are always successful. So, the transmission errors via two reservations of stream $A \to B$ correlate.

This example shows that the statistical model may overestimate $P_{\hat{\theta}}(R)$, which means that choosing $R^{new} = R^*$ may violate the PLR requirement (2). Taking this fact into account, we develop an enhancement of *the third step*: we check if the use of set R^* given by the statistical model may violate the PLR requirement and adjust the set if needed.

To check if the use of set R violates the PLR requirement, we consider $\delta_R(j) = 1 - \prod_{i \in R} (1 - X_{ij})$, which is the indicator that the j-th packet has been successfully transmitted by at least one of the reservations from set R. Then we calculate the successful packet transmission probability for reservation set R as

$$P_\delta(R) = \frac{1}{h} \sum_{j=1}^{h} \delta_R(j). \tag{6}$$

This estimate is more accurate than (5) and may be used instead of it:

$$F_{W,1-P_\delta(R)}(\lfloor W \cdot PLR_0 \rfloor) \geq \alpha. \tag{7}$$

However, (6) is undefined for $R \not\subseteq R^{cur}$, and, in this case, we obey the recommendation of the statistical model: $R^{new} \leftarrow R^*$.

Otherwise, i.e. if $R^* \subseteq R^{cur}$, we form R^{new} as follows.

- If PLR requirement (7) is not met for the current set R^{cur}, we need to establish more reservations. $R^* \subseteq R^{cur}$ means that the statistical model does not increase the number of reservations, because it overestimates the probability of successful packet transmission. So, we ignore recommendation R^* and establish an additional reservation: $R^{new} = R^{cur} \cup i'$.
- Otherwise, i.e. if PLR requirement (7) is met for current set R^{cur}, it may be possible to reduce the number of reservations. Starting from $R = R^{cur}$, we remove reservations from R one by one until $|R| = |R^*|$ or further removal of reservations breaks PLR requirement (7). Each time we select *the worst reservation* as the candidate to be removed from given set R, i.e. we select reservation $i = i_w$ which maximizes value $P_\delta(R \setminus \{i\})$.

3.4 Enhancement to Reduce Fluctuation

Even applied to the channel with a constant probability of successful packet transmission, the procedure described above may fluctuate, i.e. give various sets R_t^* of reservations at different time moments t. Let us describe how to reduce such fluctuation.

We propose to choose set R_t^{new} at moment t, based on the recommendations $R_t^*, R_{t-1}^*, \ldots, R_{t-l+1}^*$ given at moments $t, t-1, \ldots, t-l+1$, where l is a history size limit. Note that if $l = 1$ the enhancement has no effect.

We remove reservations only after several repeats of such recommendation. Strictly speaking, if $|R_t^*| < |R^{cur}|$ we may remove not more than $\min_{k=0,1,\ldots l-1} |R_{t-k}^*| - |R^{cur}|$. We remove reservations in the way described in Section 3.3.

However, if we need to add new reservations, we do it immediately: if $|R_t^*| > |R^{cur}|$ then $R^{new} \leftarrow R^*$.

3.5 Adaptive Selection of Reservation Type

In this Section, we extend the procedure to select the appropriate type of protection. For this purpose, we consider separately sets R_{oh} and R_{th} of reservations with one-hop and two-hop protection, and then apply ideas discussed in Sections 3.1 – 3.4.

The first step of the procedure is the same as in Section 3.2 except for (4). Namely, we estimate $p_{i'}$ for a new reservation i' of type y considering only the existing reservations of the same type $R_y^{cur} \subset R^{cur}$:

$$\hat{p}_{i'} = \frac{1}{|R_y^{cur}|} \sum_{i \in R_y^{cur}} \hat{p}_i. \tag{8}$$

If there is no reservation of type y, then (8) is not valid. In this case, we assume that the value of $\hat{p}_{i'}$ has not changed since the last time it was calculated. If we have no statistics for type y for z times of the procedure execution, we set $\hat{p}_{i'} = 1$ for this type of reservations to try it.

At the second step, we find the cheapest set $R^* = R_{oh}^* \cup R_{th}^*$ that satisfies (2). As reservations with two-hop protection are c_{th} times more expensive than reservations with one-hop protection, the total cost of set R^* is $C_{R^*} = |R_{oh}^*| + c_{th}|R_{th}^*|$.

At *the third step*, we improve QoS support and reduce fluctuation similarly to Sections 3.3 and 3.4.

To improve QoS support, we analyze whether the model recommendation meets PLR requirement (7) and form a new set of reservations as follows.

- Consider the case when (7) *is not met* for the current set R^{cur}. In this case, the procedure should establish new reservations but should not release existing ones. It is performed as follows.
  1. If the statistical model recommends to add at least one reservation, we add it without removing existing ones.
  2. Otherwise, i.e. if model does not recommend to add reservations, we consider that the model fails and add a new reservation. The type of this reservation is selected as follows.
     Let $\hat{p}_{oh'}$ and $\hat{p}_{th'}$ be estimates (8) for new reservations with one-hop and two-hop protection, respectively. Assuming that errors are not correlated, we can estimate the least number κ of reservations with one-hop protection that provide the same or less PLR as compared to one reservation with two-hop protection: $(1 - \hat{p}_{oh'})^\kappa \leq (1 - \hat{p}_{th'})$. If $\kappa > c_{th}$ then we select two-hop protection for the new reservation. Otherwise, we choose one-hop protection.
- Now consider the case when PLR requirement (7) *is met* for the current set R^{cur}. The model may recommend to add new reservations, to release existing reservations, or both to add reservations of one type and to release reservations of another type.

1. If the model recommends only to add some reservations, we follow this recommendation.
2. If the model recommends only to release reservations of the same type or of different types, we try to release these reservations greedily one by one, while PLR requirement (7) is met. As reservations differ by both quality and cost, each time we select the reservation with the smallest net utility $u_R(i)$:

$$u_R(i) = \frac{P_\delta(R) - P_\delta(R \setminus \{i\})}{c_i}, \tag{9}$$

   where c_i is the cost of reservation i. For reservations with one-hop protection it equals $c_{oh} = 1$, and for reservations with two-hop protection it equals c_{th}.
3. If the model recommends to remove reservations of one type and to add reservations of the other type, we, first, try to release reservations of the first type while PLR requirement (7) is met and then add reservations of the second type.

To reduce fluctuation, we perform actions similar to those described in Section 3.4, except for that we consider sets of reservations with one-hop and two-hop protection separately.

4 Performance Evaluation

4.1 Simulation Setup

To evaluate the efficiency of the proposed method, we use a simulation environment consisting of two components.

The first component is the popular network simulator ns-3 [17], which implements PHY and MAC layers of the IEEE 802.11 standard quite accurately. Using ns-3, we carry out a set of experiments, in which we observe the transmission of a CBR stream between two appointed stations by means of MCCA[2] in different scenarios. Specifically, we transmit a voice stream with packet interarrival time $T = 20$ ms corresponding to G.729 codec [18] between two stations by means of a fixed number N of reservations. In different scenarios described in detail below, transmissions of voice packets in established MCCAOPs are affected by a random noise and interference caused by transmissions of other stations. As a result, in each experiment we obtain a binary matrix $\tilde{X}_{N \times T_{exp}}$ representing statistics of all transmission attempts in N reservations with experiment duration $T_{exp} = 10^6$. Unless explicitly stated, we suppose that all time intervals are measured in time units T.

The second component of the simulation environment is the proposed method itself implemented in the R software environment [19]. The method takes matrix

[2] We have extended ns-3 with the MCCA model in [4]

$\tilde{X}$ as the input data and keeps set R^{cur} of active reservations. Note that each reservation from set R^{cur} corresponds to a particular row in $\tilde{X}$. If the method decides to establish a new reservation, it randomly chooses an unused reservation (unused row) and adds it to the R^{cur}. We assume that the stations have enough channel resources to do it. Otherwise, if the method decides to release a reservation it simply removes it from R^{cur}. At the beginning of each experiment, R^{cur} contains the only reservation randomly chosen from $\{1, .., N\}$.

To verify if the proposed method meets requirements specified in Section 2.3, we measure the following performance indices:

1. *QoS Violation Ratio (QVR)*, which is the fraction of time when the PLR requirement (1) is not met;

2. *Mean Cost of the Reservation set (MCR)*, which is the sum of the mean number of reservations with one-hop protection and two-hop protection in R^{cur} weighted by their costs c_{oh} and c_{th};

3. *Frequency of Adding (FA)*, which is the mean number of reservations with one-hop or two-hop protection, added by the method to R^{cur} during interval t_{setup}.

In all experiments presented bellow, we use the following values of parameters: $PLR^{QoS} = 0.05$, $\alpha = 0.95$, $W = 50$. We suppose that $t_{setup} = 50$, $c_{th} = 4$. The maximum number of reservations $N = 15$, which is more than enough to satisfy QoS requirements in all scenarios.

4.2 Statistical Model Adjustment

In the first series of experiments, we consider the basic procedure described in Section 3.2 and analyze the influence of the parameters τ and h on the ability of the method to react to the changes of channel conditions. For that, let us consider a scenario in which transmissions of voice packets are affected by random noise modeled as follows. We divide the time of experiment T_{exp} into intervals of equal duration T_{change}. At the beginning of each interval, we randomly choose the noise level from four possible values and keep it constant until the end of the interval. These four levels are preliminary chosen in such a way that at each particular level we need 1, 2, 3, or 4 reservations to meet the PLR requirement. So, in this scenario, T_{change} characterizes the mean rate of channel conditions changes.

We carry out an extensive evaluation of our method in a wide range of τ, h and T_{change} values. The results show that with any values of h and T_{change}, using smaller values of τ allows to decrease QVR and MCR without significant growth of FA. In other words, it is worth choosing the value of τ as low as possible. On the other hand, τ cannot be less than t_{setup} as the reservations can be established or released only after t_{setup} interval. So, further we use the minimal possible value of $\tau = t_{setup} = 50$.

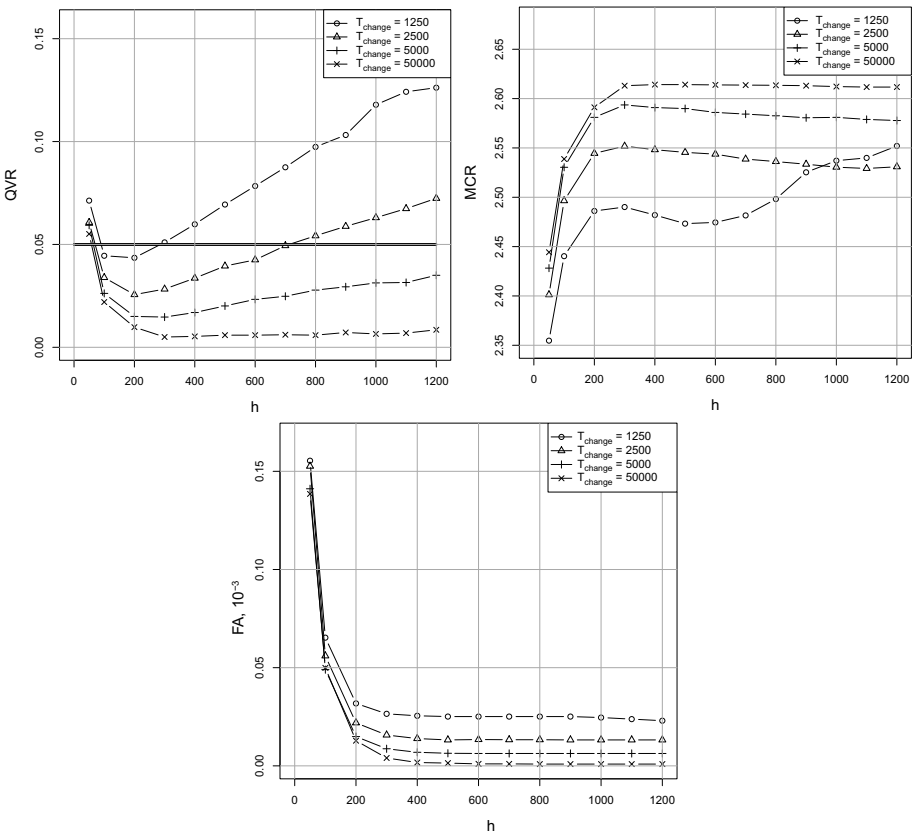


Fig. 4. Results of experiments for different values of h and T_{change}, τ=50

As for parameter h that limits the amount of statistics used to tune our model, the results[3] presented in Fig. 4 show that we should choose neither too low nor too high values. Choosing too low values of h ($h < 100$) leads to errors in estimates (3) and (4) of the model parameters, which, in turn, results in violation of the PLR requirement and high fluctuation. On the other hand, too high values of h ($h > 300$) reduce fluctuation, but do not allow to react timely to the changes of channel conditions. In particular, this occurs in the case of small values of T_{change}. Though the optimal value of h depends on the particular scenario, we recommend to use values $h \in [100, 300]$. We also check this recommendation in the following experiments.

4.3 Performance Evaluation of Enhanced Procedures

In the second series of experiments, we consider a more general scenario in which transmissions of voice packets are affected by both random noise and interference

[3] We repeat the experiments as many times as needed to achieve the error less than 5%.

with transmissions of other stations. Specifically, using ns-3 we run experiments in which we observe transmission of a voice stream between two stations A and B in the middle of a 6x5 grid network by means of a fixed number of reservations with one-hop protection. During the experiment other stations periodically start voice streams of limited duration. The source-destination pairs for these streams are chosen randomly, and on average each station is the source of one voice stream. The voice streams are transmitted with MCCA. In addition to voice streams, all stations are overloaded with background traffic which is transmitted by means of the EDCA random access method in the intervals not occupied by MCCA. Interested reader can find more detailed description of the similar ns-3 experiments (e.g. used PHY and MAC settings) in our previous paper [4].

We use the results obtained from the ns-3 experiment described above (i.e. binary matrix $\tilde{X}$) as the input data for evaluation of our method. In particular, we compare efficiency of procedures with and without enhancements introduced in Sections 3.3–3.4. The results of this comparison are presented in Fig. 5. The curve denoted as "Enhanced-l" corresponds to the procedure with both enhancements improving QoS and reducing fluctuation in use, and history size limit l. The obtained results show that the basic procedure (see curve "Basic"), which rigorously follows recommendations of the statistical model, is incapable to meet the PLR requirement (for any h, QVR is above $1 - \alpha$ threshold). The cause of such misbehavior is the assumption of the statistical model: transmission attempts in various reservations are assumed to be independent. However, this assumption does not hold in the considered scenario, because the interference from the stations outside the one-hop neighborhood of A and B causes correlated errors in different reservations. In such conditions, the statistical model tends to overestimate the successful transmission probability (5) and therefore to underestimate the required number of reservations.

The QoS improving enhancement developed in Section 3.3 allows to significantly reduce the QVR value (see curve "Enhanced-1" in Fig. 5). However, without the enhancement reducing fluctuation, the PLR requirement is still not met because of the following reason. When removing reservations, the procedure makes a decision based only on the statistics of the last h packets. So, the procedure often releases reservations and at the next execution establishes them back. As shown in Fig.5, it leads to high fluctuation (FA is even higher than for the basic procedure) and therefore violation of the PLR requirement.

When $l > 1$, both enhancements are in force. The results presented in Fig.5 confirm that the usage of the enhancement proposed in Section 3.4 allows to reduce fluctuation and, eventually, to meet the PLR requirement. However, too high values of l increase resource consumption, as they introduce an additional delay before removing reservations. So, to meet the PLR requirement, both enhancements should be used, while the actual value of l should be chosen, depending on τ and h values. For example, in the considered scenario with $\tau = 50$ and $h = 300$, we can choose $l = 10$.

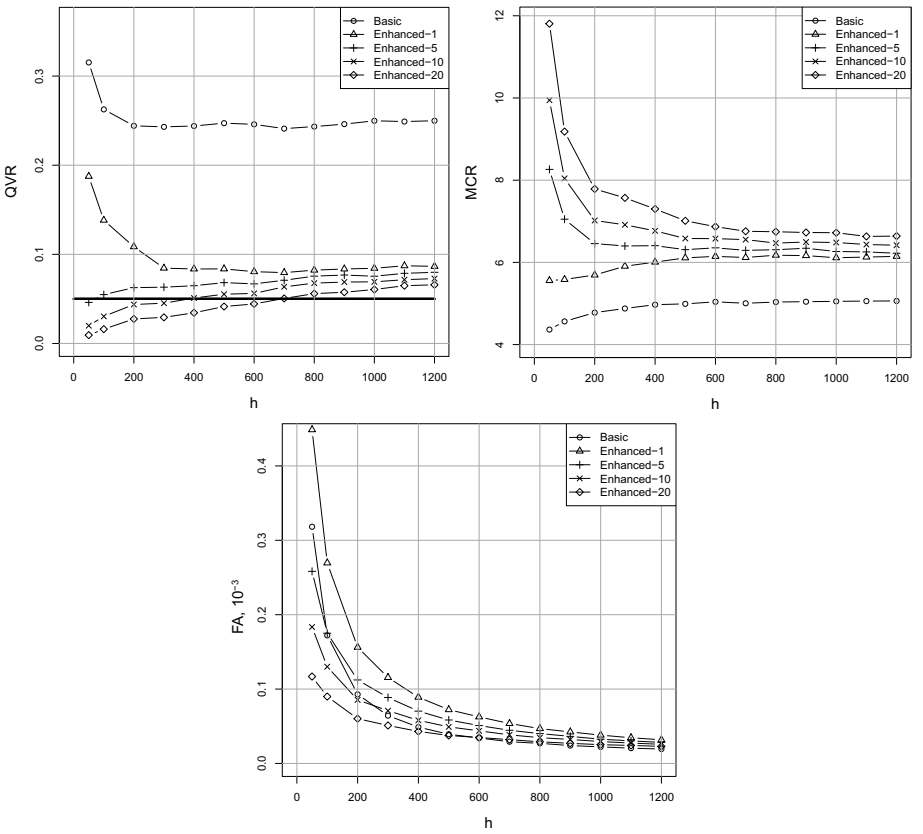


Fig. 5. Results of experiments for different procedures with and without enhancements, τ=50

4.4 Adaptive Selection of Protection Type

The results obtained in Section 4.3 also show that in case of high interference the usage of one-hop protection leads to high resource consumption. Indeed, the mean probability of successful transmission in each reservation with one-hop protection is very low (approximately 0.5), which forces us to establish at least 7 reservations to meet the PLR requirement. To decrease resource consumption, in Section 3.5 we propose the procedure which adaptively selects protection type (one-hop or two-hop) for the established reservations, taking into account the current channel conditions.

To obtain more tractable results and to evaluate the efficiency of the adaptive procedure, let us consider a simpler scenario than in Section 4.3, shown in Fig. 6. As before, during the ns-3 experiment station A transmits a voice stream to station B with a fixed number of reservations with either one-hop or two-hop protection. Station C periodically (with a fixed period T_0) starts transmission of saturated EDCA traffic to station D (e.g., initiates a TCP session) which lasts

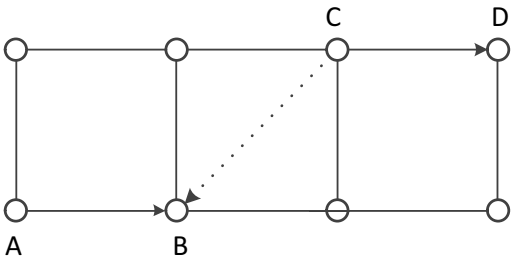


Fig. 6. Network scenario considered in Section 4.4

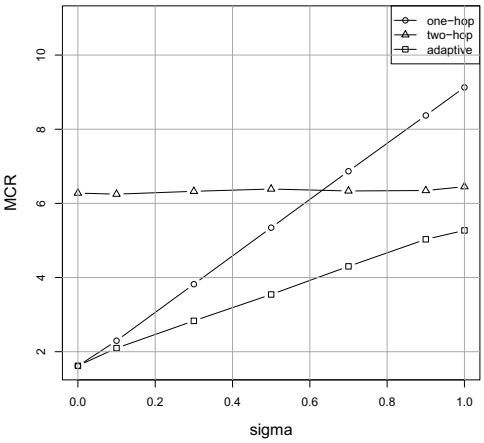


Fig. 7. Channel resource consumption vs. σ, $\tau = 50, h = 300, l = 5$

for σT_0, $\sigma \in [0, 1]$. The transmission of station C causes interference at station B. Also during the whole experiment packet transmissions are affected by random noise of a fixed level. Summing up, packet transmissions between stations A and B are affected by both random noise and interference from station C during σT_0 subinterval, and by random noise only during $(1 - \sigma)T_0$ subinterval.

In the scenario described above, we compare the efficiency of our method when for the established reservations the protection type is: i) only one-hop, ii) only two-hop, iii) chosen adaptively, as defined in Section 3.5. The results are presented in Fig.7. As the PLR requirement is always met and the fluctuation is relatively low for all σ values, we omit QVR and FA plots.

Let us explain the obtained results. When transmissions in the established reservations are affected by random noise only ($\sigma = 0$), the average number of reservations established by our method is 1.6 regardless of the used type of protection (one-hop or two-hop). However, the cost of reservations with two-hop protection is 4 times greater than that with one-hop, so it is inefficient to use two-hop protection in this case.

When reservations are affected by both random noise and interference ($\sigma = 1$), the usage of only one-hop protection leads to high resource consumption, as the probability of successful transmission in reservations with one-hop protection is very low. Two-hop protection allows to eliminate interference, so the average number of reservations with two-hop protection established by our method is the same as in case of $\sigma = 0$.

With adaptive selection of protection type in case of $\sigma = 1$, our method on average establishes 1 reservation with two-hop protection and 1.3 reservations with one-hop protection, which results in a lower cost comparing with non-adaptive schemes. Moreover, for all σ values adaptive selection of protection type provides the lowest resource consumption, comparing with a fixed protection type, which, in turn, proves the adaptive selection efficiency.

5 Conclusion

The method proposed in this paper allows to use MCCA for streaming real-time multimedia data with parameterized QoS support in presence of random noise and interference dynamically changing over time. Numerous experiments show that the method is robust in a vast range of scenarios. Apart from providing QoS support, the method minimizes channel resource consumption, taking into account the dynamic properties of the channel inherent to the real-life Wi-Fi Mesh networks, which, in turn, increases network capacity. It worth to note that the proposed method can be also applied with other technologies which use deterministic access methods to reserve channel resources, e.g. with ECMA 368 DRP.

Although the presented method is developed for the case of transmission of CBR streams, it can be extended for VBR streams (e.g. real-time video streaming). For that, in addition to estimation of channel conditions the statistical model should be able to estimate the incoming packet rate. This will be the direction of our future research.

Acknowledgement. The reported study was partially supported by the Ministry of education and science of Russian Federation, (research project No. 8766), by RFBR (research project No. 12-07-33067 mol a ved) and by FP7 FLAVIA project (No. FP7-257263).

References

1. IEEE 802.11s-2011: 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: Mesh Networking (2011)
2. Lyakhov, A., Pustogarov, I., Safonov, A., Yakimov, M.: Starvation effect study in IEEE 802.11 mesh networks. In: Proceedings of Third IEEE International Workshop on Enabling Technologies and Standards for Wireless Mesh Networking (MeshTech 2009), Macao SAR, P.R. China (2009)

3. Cicconetti, C., Lenzini, L., Mingozzi, E.: Scheduling and Dynamic Relocation for IEEE 802.11s Mesh Deterministic Access. In: Proceedings of 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2008), California, USA, pp. 19–27 (2008)
4. Krasilov, A., Lyakhov, A., Safonov, A.: Interference, even with MCCA channel access method in IEEE 802.11s mesh networks. In: Proceedings of IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2011), Valencia, Spain, pp. 752–757 (2011)
5. Khorov, E., Lyakhov, A., Safonov, A.: Flexibility of Routing Framework Architecture in IEEE 802.11 s Mesh Networks. In: Proceedings of IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2011), Valencia, Spain, pp. 777–782 (2011)
6. Shvets, E., Lyakhov, A.: Mathematical model of MCCA-based streaming process in mesh networks in the presence of noise. In: Proceedings of Wireless Communications and Networking Conference (WCNC 2012), pp. 1887–1892 (2012)
7. Yu, X., Navaratnam, P., Moessner, K.: Resource Reservation Schemes for IEEE 802.11-Based Wireless Networks: A Survey. IEEE Communications Surveys & Tutorials (99), 1–20 (2012)
8. ECMA: High Rate Ultra Wideband PHY and MAC Standard, ECMA-368 (December 2007)
9. IEEE 802.11-2007: 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 (2007)
10. Boggia, G., Camarda, P., Grieco, L., Mascolo, S.: Feedback-Based Control for Providing Real-Time Services With the 802.11e MAC. IEEE/ACM Transactions on Networking 15(2), 323–333 (2007)
11. Skyrianoglou, D., Passas, N., Salkintzis, A.K.: ARROW: An Efficient Traffic Scheduling Algorithm for IEEE 802.11e HCCA. IEEE Transactions on Wireless Communications 5(12), 3558–3567 (2006)
12. Kuo, W.-K., Wu, C.-Y.: Supporting Real-Time VBR Video Transport on WiMedia-Based Wireless Personal Area Networks. IEEE Transactions on Vehicular Technology 58(4), 1965–1971 (2009)
13. Daneshi, M., Pan, J., Ganti, S.: Towards an Efficient Reservation Algorithm for Distributed Reservation Protocols. In: Proceedings of the 29th Conference on Information Communications (INFOCOM 2010), pp. 1855–1863 (2010)
14. Shvets, E., Lyakhov, A., Safonov, A., Khorov, E.: Analytical model of IEEE 802.11s MCCA-based streaming in the presence of noise. SIGMETRICS Perform. Eval. Rev. 39(2), 38–40 (2011)
15. FLAVIA: FLexible Architecture for Virtualizable future wireless Internet Access, FP7 research project, `http://www.ict-flavia.eu/`
16. Ibragimov, I.A.: Maximum-likelihood method. Encyclopedia of Mathematics, ISBN 1-4020-0198-3
17. The ns-3 network simulator, `http://www.nsnam.org/`
18. ITU-T: Recommendation G.729 Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CSACELP). Tech. rep. (January 2007)
19. The R Project for Statistical Computing, `http://www.r-project.org/`

Distributed Clock Synchronization Algorithm for Wide-Range TDMA Ad Hoc Networks

Dmitry Doronin[1] and Denis Fakhriev[2]

[1] Moscow Institute of Physics and Technology, Telum JSC, Moscow
dmitr-doronin@yandex.ru
[2] Telum JSC, Moscow
fakhriev@telum.ru

Abstract. In this paper a new distributed clock synchronization algorithm for wireless multihop ad hoc network is presented. The algorithm is based on TSF (Timing Synchronization Function) used in IEEE 802.11. We extended TSF with determination of propagation delay between neighbors. This is especially important for wide-range networks where propagation delays can be significant. The performance analysis of the protocol is carried out with a simulation model.

Keywords: synchronization, ad hoc networks, MANET, TDMA, propagation delay, decentralized.

1 Introduction

The multihop ad hoc network is a set of nodes that can exchange data without a base node or any other central coordinator. Most papers dedicated to study on MANET deal with networks that are based on random access, e.g. IEEE 802.11 (WiFi). However, recently more and more papers that research TDMA-based networks appear. As opposed to random access networks, TDMA (Time Division Multiple Access) networks let nodes allocate time intervals for data transmission. It is especially important to provide QoS for multimedia data flows, sensitive to delays and packet losses.

The main idea of TDMA is division of time into frames which are divided into fixed quantity of time-slots. In order to get access to these time-slots node has to reserve them in advance, i.e. inform all its two-hop neighbors that it is going to use these slots. Successful completion of slot reservation procedure lets nodes transmit data without collisions with other nodes in the network (two hop reservation guarantees the absence of hidden and exposed stations). Protocols that implement slot reservation principle in TDMA MANET are, for example, USAP [1], FPRP [2], E-TDMA [3], etc.

To provide proper work of TDMA network, nodes must be synchronized, i.e. they must count time-slots simultaneously. Clock indications at nodes in the network differ because of two reasons. The first is that nodes are turned on at different moments so clocks have different offsets. The second is that clock skews also vary. Existing technology cannot produce absolutely identical quartz

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 112–124, 2013.

oscillators for clocks, so different quartzes have slightly different frequencies. Quartz clock accuracy lies between 100 ppm for cheep devices and 0.1 ppm for high-quality devices. Moreover clock skew changes over time due to quartz ageing.

Required synchronization can be obtained from Global Navigation Satellite Systems (GNSS), but usage of such system is sometimes impossible because of weak signal (in a city with dense building, indoors, in the forest, etc.). Thus TDMA networks require inherent mechanism that provides synchronization of nodes.

Existing approaches to this problem can be divided into centralized and decentralized ones.

Centralized protocols choose a master node that sends synchronization signals to other nodes in the network. To this kind of protocols belong, for example, NTP [4], PTP [5], synchronization in cellular networks (synchronization at MS and BS). But implementation of this protocols in multihop MANETs is rather complicated due to evident equality of nodes, difficulties in master election and requirement of propagation of synchronization multiple hops away from the master.

In decentralized protocols every node somehow adjusts its clock according to some algorithm, and as a result clock indications at nodes converge to some global value. The advantage of such protocols is robustness (network does not collapse in the case of failure of one or more nodes in the network) which is highly important for ad hoc networks.

In this paper we present a decentralized time synchronization protocol for MANET. The main idea is that nodes compare their clock indications and a node with smaller indication synchronizes to a node with the higher one. Significant feature of this protocol is that it allows nodes to synchronize taking propagation delays between them into account. This is especially important for long-range MANET, where distances between nodes can be large. To evaluate performance of the proposed protocol and to compare it with the protocol, described in [10], we performed simulations.

The rest of this paper is organized as follows. At first we give a brief survey of existing protocols. In section 3 we describe proposed protocol. Section 4 describes the model we used to simulate the protocol and discusses the results of the simulation.

2 Related Work

A common centralized approach for synchronization in multihop network is to organize the network into a rooted tree as in the Time-synchronization Protocol for Sensor Networks (TPSN)[6] and in the Flooding Time Synchronization Protocol (FTSP) [7]. Initially one node is elected to be the global clock reference, then a spanning tree rooted at that node is built. Afterwards, each node synchronizes itself with its parent. The main drawback of these protocols is that in

the event of root failure a new root election is initiated which leads to additional overhead and potentially long period of network de-synchronization.

The simplest decentralized algorithm is TSF (Timing Synchronization Function) used IEEE 802.11. Nodes periodically send beacons with timestamps. A node that received beacon compares its own clock indication with the beacon's timestamp. If the last is larger, the node forwards its clock to the received timestamp.

A common drawback of aforementioned protocols is that they do not take propagation delays into account. It is required for proper work of TDMA network that possible inaccuracy of synchronization was included into every slot as guard interval (otherwise, packets sent in adjacent slots can overlap leading to collisions). For short-range networks this guard is insignificant but in long-range networks guard interval can occupy a considerable portion of slot length. Thus in case of synchronization without respect to propagation delay throughput of the network can significantly reduce due to long guard intervals.

Method to determine propagation delay was at first developed in synchronization protocols for wired networks. One of such protocols is NTP. The main idea of the method is two-way exchange of packets with timestamps between nodes and recording of the reception times.

Synchronization algorithms can adjust clock indication in two ways. Node can determine the difference in clock indications with another node and just change its clock offset, like in 802.11 TSF, or can additionally compensate the difference in clock skew, like in [8]. The last mechanism allows nodes to send synchronization messages less frequently.

A method of decentralized synchronization, different from TSF, was proposed in [9]. Instead of synchronization to the fastest clock, every node, according to the algorithm, calculates adjustment multiplier for its clock indication based on received timestamps. In consequence clock indications at all nodes converge to some average indication. This protocol does not take propagation delays into account. Algorithm, described in [10], improves protocol [9] and synchronizes wireless nodes with regard for propagation delay. Similar ideas were presented in [11]. Protocols [9], [10], [11] have high accuracy but rather long convergence time.

In proposed approach we take TSF and add propagation delay determination. Developed algorithm has a good balance of performance and complexity.

3 Protocol Description

3.1 MAC Layer Functioning Assumptions

In this paper we assume that all nodes in the network operate with TDMA-based MAC protocol.

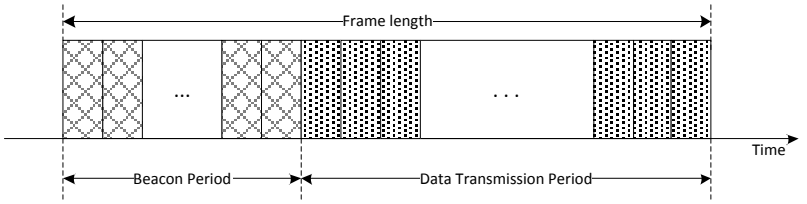


Fig. 1. TDMA channel structure

Figure 1 illustrates typical frame structure in TDMA-based MANETs. The channel time is divided into frames, with each frame composed of two parts: a beacon period (BP) and a data transmission period (DTP). A BP is consists of n beacon slots. Every slot is assigned to a unique node. Thus each node gets an opportunity to transmit its control packet (beacon) in every frame.

A DTP consists of m data slots, in which nodes may transmit user data packets. In general, nodes may get access to data slots using any random access method (e.g. ALOHA and CSMA protocols) or any deterministic access method (e.g. USAP, FPRP or E-TDMA).

Proposed protocol provides no mechanism for neighbor discovery. We assume that some other protocol exists for that purpose, i.e. there is a somehow implemented *Neighborhood management module* which informs a node about two events:

1. A new neighbor appeared;
2. A node that was considered to be a neighbor, is no longer neighbor.

3.2 Delay Determination

In this section a mechanism of propagation delay determination is described.

Propagation delay is determined by three-way beacon exchange. Assume that a node **A** wishes to determine propagation delay to its neighbor node **B**. **A** sends beacon, containing request to **B**, and records time T_1^A when the beacon was sent. **B** receives the beacon and records reception time T_2^B. At the moment T_3^B **B** sends beacon, containing reply to **A** with values T_3^B and T_2^B. **A** receives the beacon, records reception time T_4^A and calculates delay

$$delay = \frac{(T_4^A - T_1^A) - (T_3^B - T_2^B)}{2}. \tag{1}$$

Then **A** sends to **B** calculated *delay* value in beacon. After this procedure both **A** and **B** know propagation delay between them.

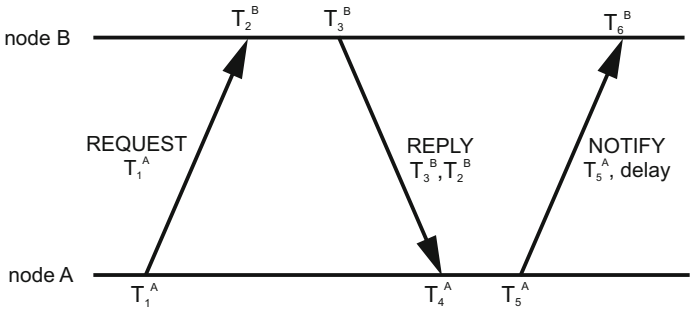


Fig. 2. Three-way beacon exchange for delay determination

3.3 Protocol Databases

A node that implements this protocol maintains five databases. Every database is a set of tuples. During protocol operation a tuple, corresponding to a neighbor, moves between databases. At any moment of time a tuple, corresponding to a neighbor node, is kept in exactly one database. A tuple can be removed from a database without being added to another database in case when neighborhood management module reports that corresponding node is no longer a neighbor. Databases are:

Neighbor Set stores data about neighbors, the node is going to measure delay to.

Wait For Reply Set stores data about neighbors, the node has sent requests for delay measurement.

Reply Set stores data about neighbors, the node has received requests from.

Notify Set stores data about neighbors, the node has to send calculated delay to.

Wait For Notify Set stores data about neighbors from which the node waits for beacon, containing calculated delay.

Every tuple in any database has a field that contains address of corresponding node.

3.4 Beacon Structure

Depending on the information transmitted in a beacon, the last can belong to one of the following types:

1. *REQUEST* – delay determination request. Contains T_1 – time when request was sent;
2. *REPLY* – reply to delay_request. Contains T_2 – time when request was received, and T_3 – time when reply was sent;
3. *NOTIFY* – beacon contains calculated delay. Contains T_5 – time when the beacon was sent and *delay* – calculated propagation delay between two nodes;
4. *EMPTY* – the beacon is not used in delay determination.

All beacons except of type *EMPTY* contain address *SrcAddr* of a node that sent the beacon and address *DstAddr* of a node, the beacon is destined to.

3.5 Protocol Operation

When a node **A** decides to measure delay to its neighbor **B**, it moves corresponding tuple from *Neighbor Set* to *Wait For Reply Set* and sends $REQUEST$ to **B**. When **B** receives $REQUEST$, it moves tuple, corresponding to **A**, from *Neighbor Set* to *Reply Set* and records reception time to that tuple. Node **B** at the beginning of its beacon interval pops tuple from *Reply Set* and sends $REPLY$ to **A**. After that the tuple is moved from *Reply Set* to *Wait For Notify Set*. node **A** receives $REPLY$, extracts timestamp T_3^B and calculates propagation delay t_p. node **A** compares its clock indication T_{rec}^A with $T_3^B + t_p$ (this is clock indication at node **B** when the beacon was received by node **A**). If $T_3^B + t_p > T_{rec}^A$ node **A** sets its clock indication to $T_3^B + t_p$. If $T_3^B + t_p < T_{rec}^A$ node **A** does not change its clock indication. After that node **A** moves corresponding tuple from *Wait For Reply Set* to *Notify Set*. Node **A** at the beginning of its beacon interval pops tuple from *Notify Set*, sends $NOTIFY$ to **B** and moves the tuple to *Neighbor Set*. Node **B** receives $NOTIFY$ and extracts timestamp T_5^A and propagation delay t_p. Then node **B** compares its clock indication T_{rec}^B with $T_5^A + t_p$. If $T_5^A + t_p > T_{rec}^B$ node **A** sets its clock indication to $T_5^A + t_p$. If $T_5^A + t_p < T_{rec}^B$ node **A** does not change its clock indication. After that the tuple, corresponding to **A** is moved from *Wait For Notify Set* to *Neighbor Set*.

A node between two sequential beacon intervals can receive requests from more than one nodes. With help of aforementioned databases a node can sequentially reply to different nodes in different beacon intervals. At the beginning of its beacon interval node pops tuple from *Notify Set* and sends $NOTIFY$. If *Notify Set* is empty, nodes pops tuple from *Reply Set* and sends $REPLY$. If *Reply Set* is empty, node pops tuple from *Neighbor Set* and sends $REQUEST$. If *Neighbor Set* is empty node sends $EMPTY$.

Since beacons can be dropped, when a tuple is moved to a database, different from *Neighbor Set*, the tuple is added with an *obsolete_time* field. This field contains the time when the tuple will be moved to *Neighbor Set*.

4 Performance Analysis

In this section, we analyze performance of the proposed synchronization algorithm with help of a computer model, implementing. Performance is evaluated in terms of overhead, convergence time and synchronization accuracy. Algorithm is also compared with the one proposed in [10] (we will call it further FDCS – Fully Distributed Clock Synchronization). The impact of network density, clock skew inaccuracy and packet loss ratio is studied.

This section is divided as follows. The topology and model assumptions are first presented. Then we discuss protocol overhead, protocol convergence time and synchronization accuracy.

4.1 Model and Scenarios

In the scenarios studied here, the network is composed of 80 nodes uniformly distributed across a square area. Two topologies were studied: a dense one, where every node has 40 neighbors on average, and a sparse one, where every node has 4 neighbors on average. An average distance between neighbors is 10 km. Network topology is static. Nodes implement channel access model, described in section 3.1 where every frame is 0.1 second long.

Indication of the clock at node number i at time t is given by the formula

$$T_i(t) = \alpha_i t + \beta_i \tag{2}$$

where α_i is a clock skew, β_i is initial offset. α_i is uniformly distributed on the interval $[1 - \frac{\Delta\alpha}{2}, 1 + \frac{\Delta\alpha}{2}]$. β_i varies between 0 and 100 μs.

4.2 Overhead

Each node in the network is configured so that it performs three-way beacon exchange with every neighbor and synchronizes with it once in a second. Table 1 shows the global overhead due to synchronization protocol operation.

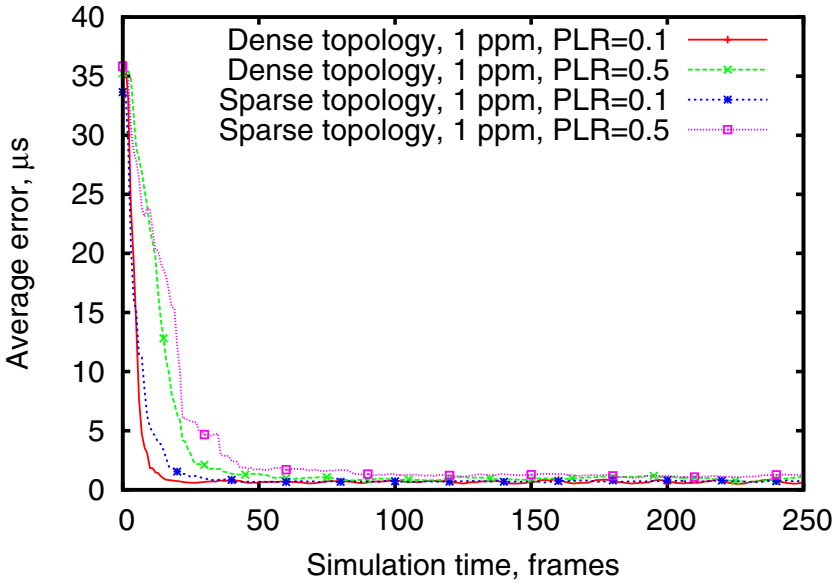
Table 1. Synchronization protocol overhead

Average number of neighbors	Overhead (kbits/s)
4	35
40	45

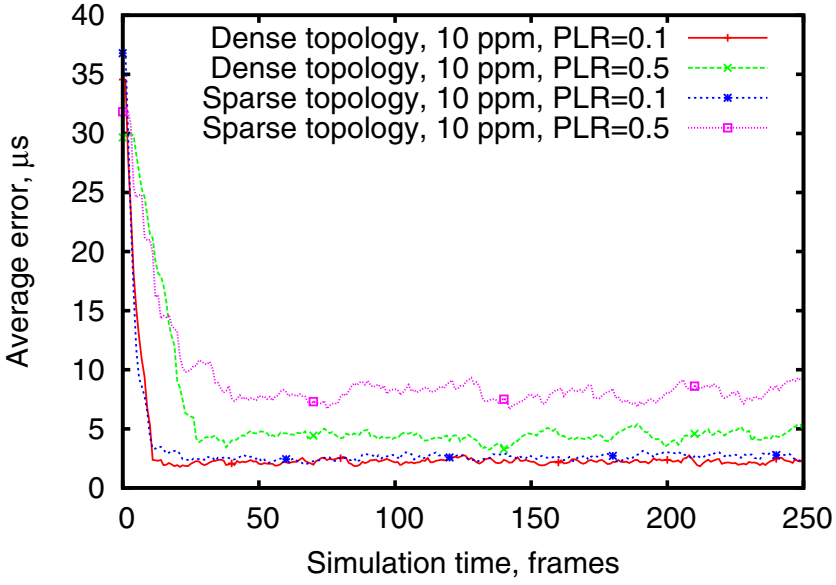
4.3 Convergence Time

Fig. 3a, 3b illustrate the average clock offset of all nodes with there respective 1-hop neighborhood (i.e. average synchronization error between directly communicating nodes) in the network where proposed protocol operates. The figures are plotted for dense and sparse topologies and different values of packet loss ratios and $\Delta\alpha$. It can be seen that the convergence time of the protocol is small and is about 40 frames or 4 seconds. The protocol operates equally well in both dense and sparse networks and convergence time does not depend on PLR and $\Delta\alpha$. In steady-state average synchronization error of proposed protocol is about 1 us.

Fig. 4a, 4b illustrates average synchronization error in the network where FDCS operates. It is seen that convergence time of FDCS is 1-2 orders greater than convergence time of proposed algorithm. This can be explained by the fact that nodes do not directly offset their clock indications but slightly change adjustment multiplier. Moreover, convergence time of FDCS depends on topology and PLR.

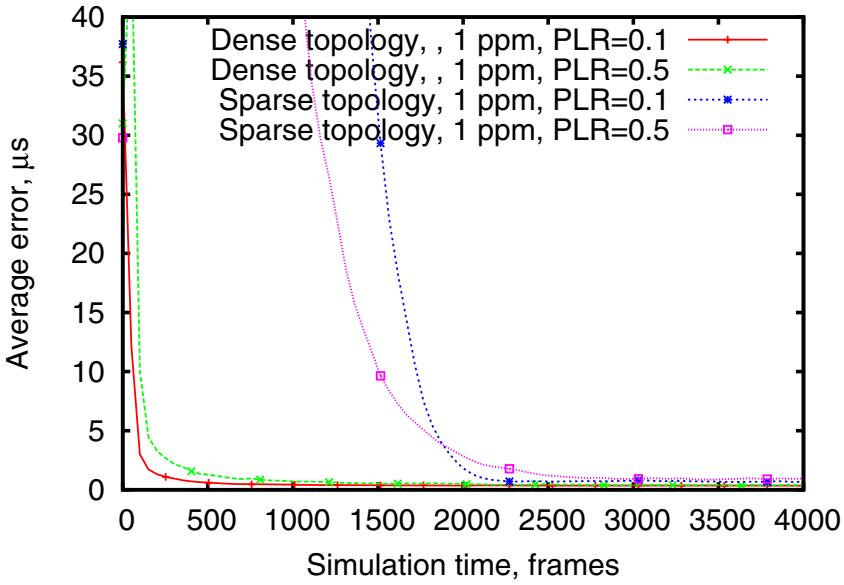


(a) 1 ppm

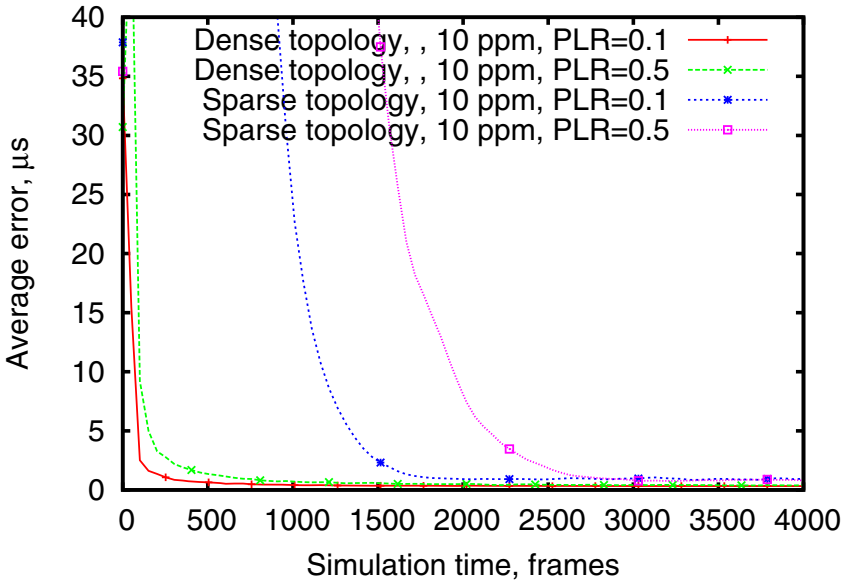


(b) 10 ppm

Fig. 3. Average synchronization error of proposed algorithm

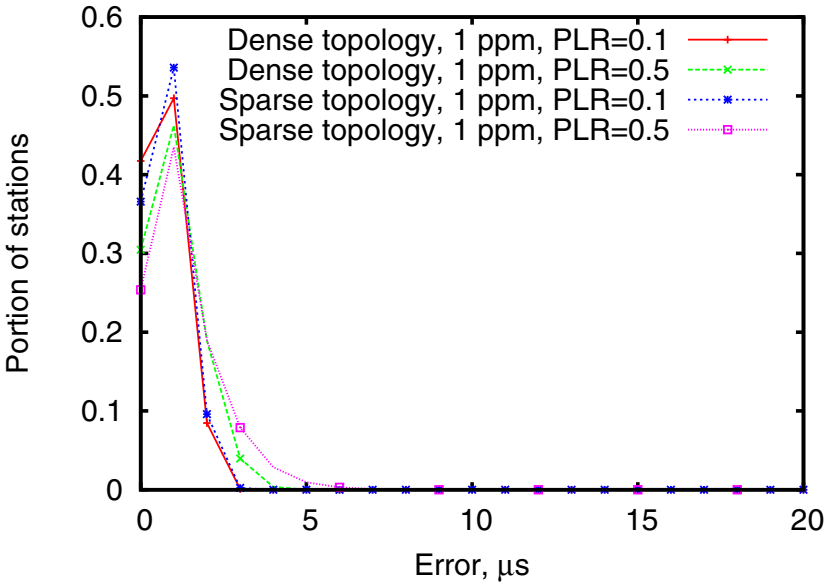


(a) 1 ppm

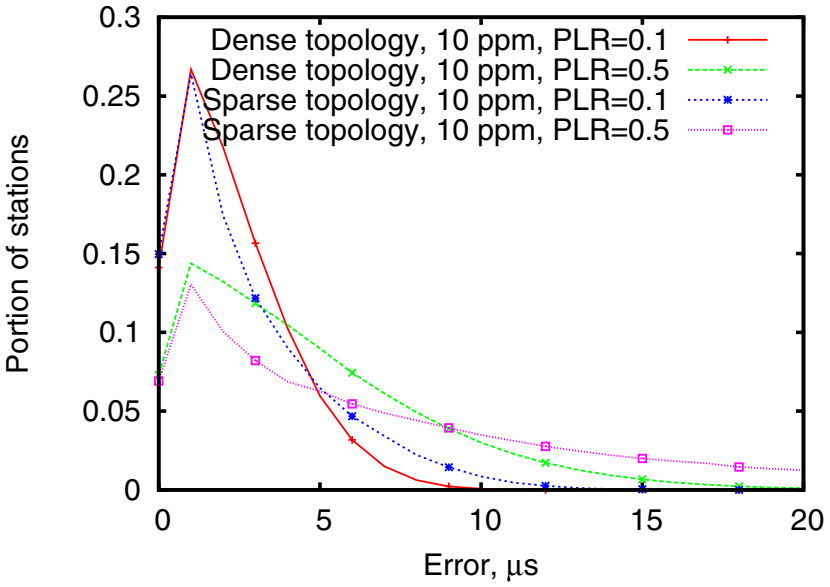


(b) 10 ppm

Fig. 4. Average synchronization error of FDCS

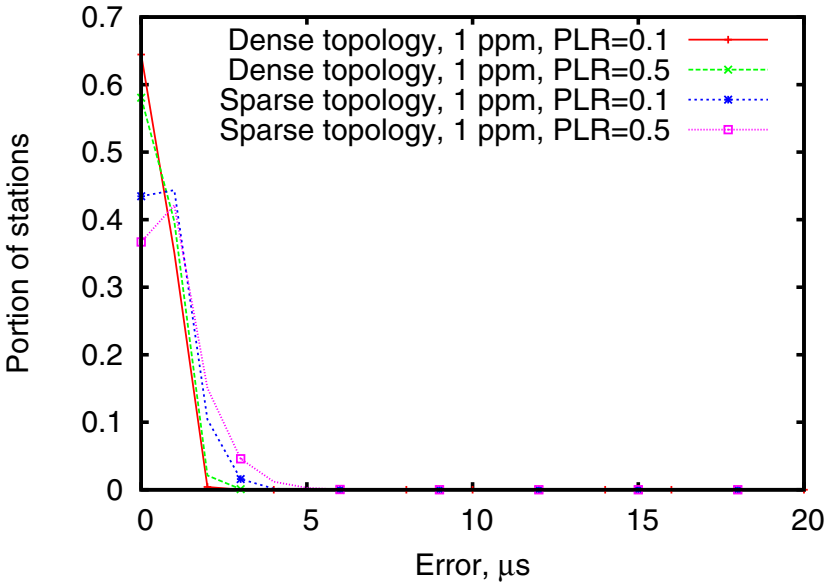


(a) 1 ppm



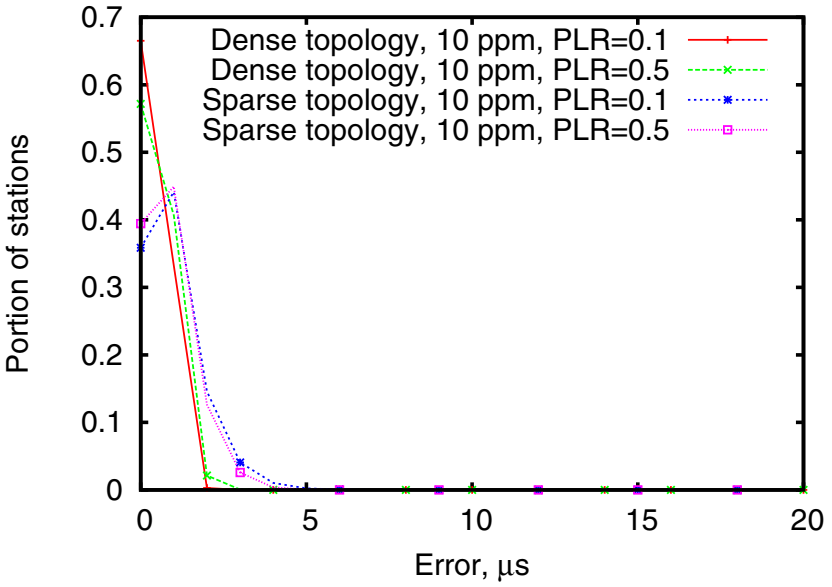
(b) 10 ppm

Fig. 5. Synchronization error distribution for proposed protocol



!p

(a) 1 ppm



(b) 10 ppm

Fig. 6. Synchronization error distribution for FDCS

4.4 Protocol Accuracy in Steady-State

In this section we analyze protocol operation in steady-state. For that we obtain the distribution of synchronization error between directly communicating nodes in steady-state, i.e. when average synchronization error fluctuates near its least value. As can be seen in fig. 3a, 3b, 4a, 4b steady-state starts after 40th frame for the proposed protocol and after 3000th frame for FDCS. Than we average out these distributions over the whole steady-state duration. Fig. 5a, 5b illustrate average distribution of synchronization error for proposed protocol and fig. 6a, 6b illustrate the same for FDCS. It can be seen that in case nodes have clock with $\Delta\alpha = 1ppm$ both proposed protocol and FDCS yield equal accuracy which is about 5 us. In case with $\Delta\alpha = 10ppm$ accuracy of FDCS stays the same, whereas accuracy of proposed algorithm becomes worse and deteriorates with increase of PLR. That can be explained by the fact that FDCS instead of just offsetting clocks compensates differences in clock skews.

5 Conclusion

A new decentralized synchronization algorithm for TDMA ad hoc networks is presented in this paper. Its main idea is that node synchronizes to its neighbor if the last has higher clock indication. Synchronization is made with regard to propagation delay. The performance analysis of the protocol and comparison with FDCS, described in [10], are carried out with help of a simulation model. The results show that proposed protocol can yield high synchronization accuracy (about 1 us) in the network where clock skew variation is less than 1 ppm. Although proposed protocol is inferior to FDCS in terms of accuracy in the case with less accurate clock, proposed algorithm converges much faster. The main advantages of the proposed synchronization protocol are: small convergence time, high synchronization accuracy in case of accurate clock, and low overhead requirements.

References

1. Young, C.D.: USAP multiple access: dynamic resource allocation for mobile multihop multichannel wireless networking. In: IEEE Military Communications Conference Proceedings, MILCOM 1999, vol. 1, pp. 271–275 (1999)
2. Zhu, C., Corson, S.S.: A five-phase reservation protocol (FPRP) for mobile ad hoc networks. In: Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 1998, March 29-April 2, vol. 1, pp. 322–331. IEEE (1998)
3. Kay, S.E.: E-TDMA: high capacity digital cellular radio. In: IEEE International Conference on Communications, ICC 1992, Conference record, SUPERCOMM/ICC 1992, Discovering a New World of Communications, June 14-18, vol. 4, pp. 1833–1835 (1992)
4. RFC 5905 Network Time Protocol Version 4: Protocol and Algorithms Specification (June 2010)

5. IEEE 1588 Standard for A Precision Clock Synchronization Protocol for Networked Measurement and Control Systems
6. Ganeriwal, S., Kumar, R., Srivastava, M.: Timingsync protocol for sensor networks. In: Proceedings of SenSys 2003 (2003)
7. Maróti, M., Kusy, B., Simon, G., Ldeczi, A.: The flooding time synchronization protocol. In: Proceedings of International Conference on Embedded Networked Sensor Systems, SenSys 2004, pp. 39–49 (2004)
8. Sheu, J.-P., Chao, C.-M., Hu, W.-K., Sun, C.-W.: A clock synchronization algorithm for multihop wireless ad hoc networks. In: Proceedings of 24th International Conference on Distributed Computing Systems, pp. 574–581 (2004)
9. Rentel, C.H., Kunz, T.: A Clock-sampling Mutual Network Time-Synchronization Algorithm for Wireless Ad Hoc Networks. In: 2005 IEEE Wireless Communications and Networking Conference, vol. 1, pp. 638–644 (2005)
10. Allard, G., Genc, V., Yelloz, J.: Fully Distributed Clock Synchronization in Wide-Range TDMA Ad Hoc Networks. In: Military Communications Conference - MILCOM 2011, November 7-10, pp. 920–925 (2011)
11. Schenato, L., Fiorentin, F.: Average TimeSynch: a consensus-based protocol for clock synchronization in wireless sensor networks. Automatica 47(9), 1878–1886 (2011)

Joint Integrated Spectrum Handoff Management and Routing in CR-MANETs: An Analytical Modeling

S. Nejatian[1], S.K. Syed-Yusof [1], N.M. Abdul Latiff[1],
V. Asadpour[2], and N. Fisal[1]

[1] Faculty of Electrical Engineering University Technology Malaysia Skudai, Malaysia
Samad.nejatian@fkegraduate.utm.my
[2] Electrical Engineering Department, University of North Dakota, USA

Abstract. The focus of this paper is on the modeling of joint Integrated Spectrum Handoff MANagement (ISHMAN) and routing in cognitive radio mobile ad hoc networks (CR-MANETs). To model the ISHMAN, we must consider three key factors: (1) channel availability and spectrum usage behaviors of the primary users (PUs); (2) the movements and the motility of the secondary users (SUs); and (3) channel quality degradation. In this paper, we propose a new model for spectrum handoff in CR-MANETs considering the above mentioned factors characterizing the united ISHMAN and routing in these networks.

Keywords: Cognitive Radio, Spectrum Mobility, Spectrum handoff, Handoff Management.

1 Introduction

In CR networks, the availability of the spectrum bands is arbitrary because of the random appearance of PUs as well as unpredictable SUs' mobility. When the PU reclaims its licensed band, which is preoccupied by the SU, the ongoing data transmission of CR user is transferred to another free spectrum band. The transferring of the SUs' operation frequency to another unused spectrum band is refereed to spectrum handoff. Therefore, the Spectrum handoff occurs when the conditions of the current channel cannot meet QoS or when a PU appears in the preoccupied licensed band. There is also a key relationship between spectrum handoff and routing In CR-MANETs. There must be a strong interaction between spectrum handoff and routing protocol in the CR-MANETs in order to avoid link failure. The established route at the network layer must not lead to any undesirable effect on PU's activities.

In this paper, we analytically model the joint ISHMAN and routing in the CR-MANETs. We also use the Markov chains to illustrate the effect of ISHMAN on the probability of spectrum handoff blocking. To the best of our knowledge, an analytical model for ISHMAN has rarely been seen in the literature.

The rest of this paper is as follows. In section 2, we describe the related works for spectrum handoff management. In section 3, we propose a unified modeling and

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 125–137, 2013.

characterization of channel availability in CR-MANETs. In section 4, we propose an analytical model for spectrum mobility and spectrum handoff in CR-MANETs. Section 5 shows the analytical model of the integrated spectrum handoff management. In section 6, the results and discussion are elaborated. Finally, Section 7 concludes the paper and presents the future work.

2 Related Works

Mobility function is a critical function in CR networks, which depends on the various parameters of the network like channel capacity, connectivity, and coverage [1]. Spectrum handoff management is particularly challenging in CR networks because of the randomness PU activity. It is more challenging in ad hoc networks due to the lack of a central body for managing and controlling the spectrum handoff procedure.

There have been a few works related to spectrum handoff management in CR-MANETs. In [2], a spectrum handoff decision making system is proposed, which uses two fuzzy logic controllers. Each SU calculates the distances between itself and all of the PUs which are active in its neighbouring using the first fuzzy logic controller. The second controller decides whether the SU has to do spectrum handoff or not. There exist some cases in which the SU can avoid doing spectrum handoff by a fair modification of its transmission power. In [3], a handoff management strategy is proposed. This scheme determines the optimal spectrum band based on a multi criteria decision making strategy, which considers the estimated transmission time, the PU presence probability, and spectrum availability time. The authors use a cooperative scheme for spectrum sensing in order to determine the spectrum availability. They also use a geolocation strategy to consider spectrum handoff in the space domain. The simulation results indicate that the proposed spectrum handoff outperforms conventional methods in terms of spectrum handoff delay and transmission efficiency. The authors of [4-5] proposed a proactive spectrum handoff scheme which is based on the statistics of channel utilization. The network coordination issue is solved without using common control channel. The collision among SUs is also deleted using a distributed channel allocation scheme. In [6], the authors have proposed integrated handoff management in CR-MANETs for the first time. They have mentioned the factors and types of mobility, which necessitate integrated mobility and handoff management in CR-MANETs. In this paper, the concept of proposed integrated handoff management in CR-MANET along with a conceptual framework is proposed. The necessary connections and related handoff algorithm are also illustrated in this paper. The authors of [7] have characterized and formulated the availability of spectrum bands in CR-MANETs. They explained and integrated the effects of various events on the spectrum holes availability in CR-MANETs using an analytical model. This integration is necessary to achieve integrated mobility and handoff management.

3 Channel Availability in CR-MANETs

In [6], the authors have considered an established route in CR-MANET from a source node S, to destination node D. They introduced two different scenarios, which lead to the handoff initiation in this route. Considering these two events, which are node mobility and spectrum mobility, the authors have introduced a conceptual model for integrated mobility and handoff management in CR-MANETs. They have also explained the model requirements, design consideration, and the algorithm for proposed handoff management.

The authors of [7] have considered the effect of spectrum heterogeneity on the probability of channel availability in CR-MANETs rather than the SUs' mobility and spectrum mobility. In a heterogeneous network, each channel experiences different levels of PU activities, and different transmission range. The authors have supposed that there is the total number of C channels in a heterogeneous network, which are classified into L types according to their different transmission ranges. Different channels occupy different spectrum bands. The number of available channels of each type at each node is c_l in which l= {1, 2,…, L}. It means that $c=c_1+c_2+\ldots+c_L$. The transmission range of channels of type l is R_l. In the initial condition when the nodes are considered fix, different channels will have different transmission range. Channel with lower frequency range needs lower transmission power. Thus, in a heterogeneous network with different channel transmission ranges, the distance between SUs must be considered in the probability of channel availability. Considering this channel classification; channel quality degradation was modeled. Fig. 1 shows this spectrum pool-based heterogeneous CR-MANET scheme. Table 1 shows the notations used in this paper. A pair of SUs transmitter-receiver, who transmit and receive respectively, can use a channel of type T_l for communication when their distance is less than R_l. When the SUs are moving, and their distance exceeds R_l, the communicating nodes must change and choose another channel. In this case, the required channel must have a transmission range longer than R_l (Fig. 2). Based on [7], the $P_{car,c}$, which is defined as the probability that there is at least one common channel among all hops in a route, considering PU's activity, SU's mobility, and channel quality degradation is defined as in (1).

$$P_{car,c}=\left[\sum_{i=1}^{L}\frac{\exp(-\lambda\pi R_{i-1}{}^2/2)-\exp(-\lambda\pi R_i{}^2/2)}{1-e^{-\frac{N}{2}}}\left(1-\prod_{k=1}^{c-\sum_{j=0}^{i-1}c_j}\left(1-\left(\frac{\alpha_k}{\alpha_k+\beta_k}\right)^2\right)\right)\right]^{n-1} \quad (1)$$

In (1), the PU's activity is considered as alternating renewal two state birth-death processes with a death rate α and birth rate β.

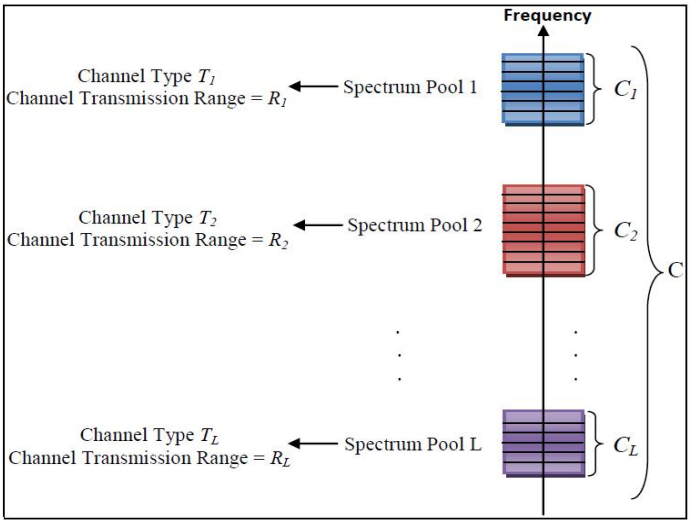


Fig. 1. Spectrum pool based heterogeneous CR-MANET scheme

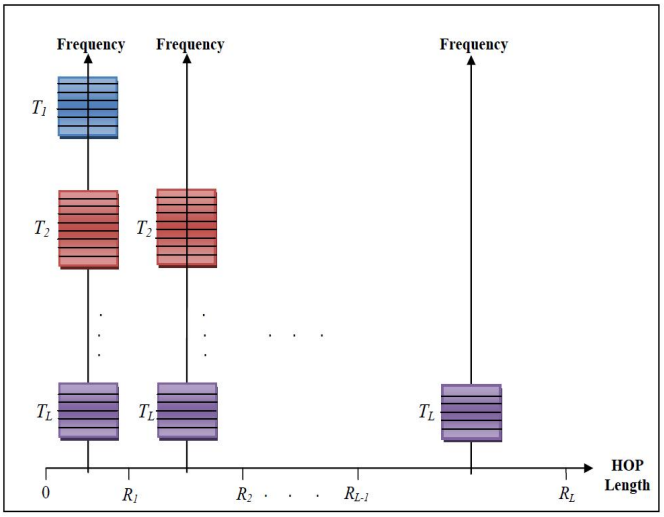


Fig. 2. Choosing the spectrum bands for communication based on the hop length

4 Analytical Modeling of Mobility and Spectrum Handoff

In this part, we exploit Markov chains to model the spectrum handoff mechanism in CR-MANETs based on the channel availability modeling above. The spectrum heterogeneity in terms of PER is also considered. The proposed integrated analytical scheme includes different mobility events in CR-MANETs such as spectrum and user mobility. It also considers the channel quality degradation, and topology variations. Once the channel quality declines or PER increases, the probability of successful

packet transmission rate is decreased. The SU detects this deterioration throughput QoS and decides to change the channel to achieve a better throughput performance Considering $P_E{}^i$ as the PER of channel of type i and equation (1), the probability of successful packet routing in a route or between n nodes is as in (2).

$$p_{spr,c} = \left[\sum_{i=1}^{L} P_{car,c}(1 - P_E{}^i) \right] =$$

$$\left[\sum_{i=1}^{L} (1 - P_E{}^i) \times \left[\frac{\exp(-\lambda\pi R^2{}_{i-1}/2) - \exp(-\lambda\pi R^2{}_i/2)}{1 - e^{-\frac{N}{2}}} \right] \times \left[1 - \prod_{k=1}^{i-1} (1 - (\frac{\alpha_k}{\alpha_k + \beta_k})^2) \right]^{c - \sum_{j=0}^{i-1} c_j} \right]^{n-1} \quad (2)$$

Table 1. Symbols used in this paper and their definitions

Symbol	Meaning
R_T	Node transmission range
C	Total number of available channel
c_l	Number of detected channels of each type at each node
l	Channel type
L	Total number of channel type
c	Total number of detected channels of different types at each node
p	Probability of a particular channel availability at each node
P_{cat,T_l}	Probability that there is at least one channel of type l among c channels between two nodes
$P_{car,c}$	Probability that there is at least one common channel among all hops in a route
λ	Poisson density of nodes' spatial distribution in the network
R_l	Transmission range of channel of type l
n	Total number of nodes in a route
N_N	Total number of the nodes in the network
A_N	Network area
P_{hb}	Probability of spectrum handoff blocking
P_{usrr}	Probability of unsuccessful rerouting
$P_{hb,sh}$	Handoff blocking probability in scheme deploying only spectrum handoff
$P_{hb,lh}$	Handoff blocking probability in scheme deploying integrated local routing and spectrum handoff management

The probability of successful packet transmission in a hop or between two nodes is also found as

$$p_{spt,c} = p_{spr,c}\Big|_{n=2} \quad (3)$$

The parameter $P_E{}^i$ shows the rate of failed packets sent because of the variable channel conditions caused by factors such as fading and shadowing. The probability of unsuccessful packet transmission in a hop or between two nodes is also found:

$$p_{uspt,c} = 1 - p_{spt,c} \tag{4}$$

The main objective is to calculate the probability distribution of spectrum handoff and also model the spectrum handoff initiation in CR-MANETs. We define the $D_{l\text{-}1\ l}$ as the case in which $R_{l\text{-}1} < d < R_l$, where d is the length of the hop. **P**, which is the Markov matrix, and is written as follows:

$$\mathbf{P} = \begin{bmatrix} P_{T_{11}} & P_{T_{12}} & \cdots & P_{T_{1L}} \\ . & . & & . \\ P_{T_{L1}} & P_{T_{L2}} & \cdots & P_{T_{LL}} \end{bmatrix}$$

Fig. 3 shows the different Markov chains for spectrum handoff modeling based on the length of the hops. The Markov chain has different states based on the different cases for $D_{l\text{-}1\ l}$. As shown in Fig. 2, when the length of the hop is less than R_1, or when the spectrum handoff occurs in the case D_{01}, the two nodes involved in the current hop can select one of the available channels of any L types. In the case D_{01}, the Markov chain is as shown in Fig. 3(a). In the case D_{12}, the nodes involved in spectrum handoff can select one channel among available channels from type k, in which $k \neq 1$, as shown in Fig. 3(b). The other situations can be determined based on the claims above.Suppose that two nodes are communicating in a channel of type k, they can continue communicating in the current spectrum pool under two different conditions. These two conditions, in the case of $D_{i\text{-}1\ i}$, are as follows:

- The packet transmission is successful in the current channel of type k.
- The packet transmission is not successful in the current channel of type k but successful on another channel of only spectrum pool k.

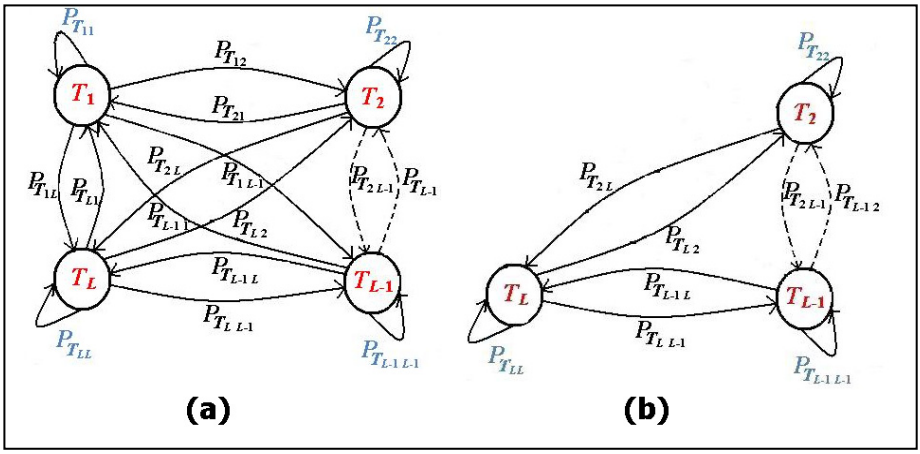


Fig. 3. Markov chains for spectrum handoff modeling based on the distance between SUs or the length of the hops

Thus, the probability of packet transmission $P_{T_{kk}}$ is calculated as follows:

$$P_{T_{kk},D_{i-1\,i}} = p_{spt,c_k} + \prod_{j=1,j\neq k}^{L} p_{uspt,c_j} \tag{5}$$

Based on the proposed claims about different Markov chains, there are two conditions where two nodes, which are communicating on a channel of type e, switch their communicating channel type to another channel of type k such that $k \in \{1,2,\ldots,L\}$, $k \neq e$. These two conditions, in the case of $D_{i\text{-}1\,i}$, are as follows:

- Unsuccessful packet transmission in a channel of type e, but successful transmission in only channel of type k.
- Unsuccessful packet transmission in a channel of type e, but successful transmission in channel type sets:

$$\mathrm{M}_i \subseteq T, \left|\mathrm{M}_i\right| = L_i \leq L \tag{6}$$

Based on the above Markov chains, there are many possible channel type sets for M_i. The channel of type j can be chosen by an identical probability among the available channel types in the set of M_i. Based on the explanation above, when the probability of channel type changing is $p_{c_{ek}}$, the probability of packet transmission is as follows:

$$P_{T_{ek},D_{i-1i}} = p_{uspt,c_e} \times \left[p_{spt,c_k} \left(\prod_{j=1,j\neq e,k}^{L} p_{uspt,c_j} \right) \right] + \left[p_{uspt,c_e} \sum_{\mathbf{M}_i \subseteq \mathbf{T}} \frac{\prod_{j=1,2,\ldots,i,T_j \in \mathbf{M}_i, j\neq e} p_{\sup,c_j} \prod_{j=1,2,\ldots,i,T_j \notin \mathbf{M}_i, j\neq e} p_{uspt,c_j}}{\left|\mathbf{M}_i\right|} \right] \tag{7}$$

The row vector $\pi^{\mathbf{P}}_{D_{l-1\,l}}$, which is composed of $\pi^{\mathrm{P}}_{D_{l-1l}}(T_i)$, demonstrates the steady state probability for $\mathbf{P}_{D_{l-1l}}$ considering different hop lengths. The value of $\pi^{\mathrm{P}}_{D_{l-1l}}(T_i)$ is calculated using the following equations:

$$\pi^{\mathbf{P}}_{D_{l-1l}} \mathbf{P}_{D_{l-1l}} = \pi^{\mathbf{P}}_{D_{l-1l}}, \sum_{i=1}^{L} \pi^{\mathrm{P}}_{D_{l-1l}}(T_i) = 1 \tag{8}$$

Ultimately, the steady state probabilities for various hop lengths for channel type i are calculated as below:

$$\pi^{\mathrm{P}}(T_i) = \sum_{l=1}^{L} \Pr(R_{l-1} < d < R_l)\, \pi^{\mathrm{P}}_{D_{l-1l}}(T_i) \tag{9}$$

where $\Pr(R_{l-1} < d < R_l)$ is defined as [7]:

$$\Pr(R_{l-1} < d < R_l) = \left[\frac{1 - \exp(-\lambda\pi {R_l}^2 / 2)}{1 - e^{-\frac{N}{2}}} \right] - \left[\frac{1 - \exp(-\lambda\pi {R_{l-1}}^2 / 2)}{1 - e^{-\frac{N}{2}}} \right]$$
$$= \left[\frac{\exp(-\lambda\pi {R_{l-1}}^2 / 2) - \exp(-\lambda\pi {R_l}^2 / 2)}{1 - e^{-\frac{N}{2}}} \right] \quad (10)$$

$$, 0 < R_{l-1} < R_l < R_T \ \& \ R_0 = 0$$

5 Integrated Mobility and Spectrum Handoff Management

To propose the integrated spectrum handoff management, we introduce the different scenarios which cause spectrum handoff initiation through an established route in a CR-MANET.

In CR-MANETs, the available spectrum bands vary over time. On the other hand, during the movement of an intermediate node, which is a member of an active route, the route may be broken and has to be repaired. To avoid route breaking, the local routing must be efficient enough. Suppose that, based on Fig. 4, a route from a source node S to the destination node D has been established. There are three different scenarios that initiate the spectrum handoff in this route, which are as follows:

SU Mobility

According to Fig 4(a), link failure happens when either node B or node F moves such that there is no channel that can support their data transmissions. Before the link breaks, rerouting must be performed. Fig. 4(b) shows that a local routing can be from node A to node E and finally joining node F. The finding of the new route and channel requires a common channel between node A, E and node F which is within the transmission range limit. In this scenario, only the links and channels between A and F will be changed, and the unified routing and spectrum handoff management system tries to solve the problem by finding a node in within the neighboring area of the damaged links.

To perform local routing, a certain amount of overlapping of the transmission range between node A, node F and the node that will take responsibility for routing the packets is necessary. To consider these two nodes as a hop counts, the maximum overlapping between these two nodes is shown in Fig. 5. The maximum overlapping area is $1.23{R_T}^2$. In a topology where nodes are uniformly distributed over the network area, the following condition guarantees the presence of at least η nodes in the overlapping area [8]:

$$N_n \geq \frac{\eta A_N}{1.23{R_T}^2} \quad (11)$$

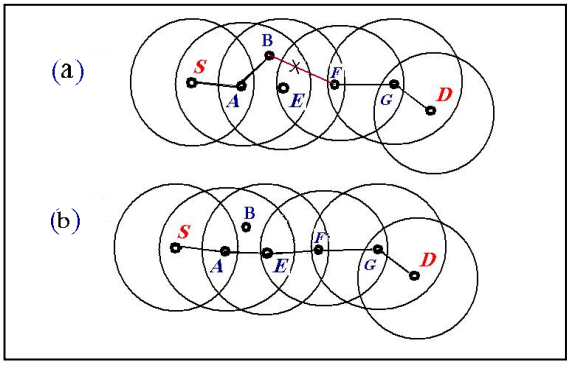


Fig. 4. Node mobility and local rerouting

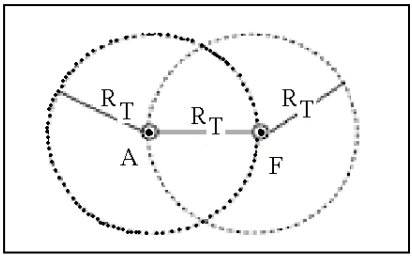


Fig. 5. Maximum overlapping between nodes A and F

To guarantee the overlapping of the transmission range of node A and node F, the following requirement must be satisfied:

$$N_n > \frac{A_N}{R_T^2} \tag{12}$$

This equation is in accordance with (11).

PU Activity Dominates

This problem happens when PU starts its activity or node E, which is a member of an active route, enters the activity area of the PU during its movement. Before the link breaks, node E must perform rerouting. In this scenario, the node E can perform spectrum handoff considering common channel availability, or it can do the local rerouting considering the mentioned conditions in the first scenario.

Spectrum Heterogeneity and Different Channel Transmission Range

The mobility of the CR user can also lead to spectrum handoff because of spectrum heterogeneity and different channel transmission ranges. Suppose that two nodes, with a distance less than R_l, are involved in an active route. Also suppose that these nodes are using a channel of type l for their data transmission. When they are moving, and their distance exceeds the R_l, they must change their communicating channel to a channel with a transmission range longer than R_l.

Consider the possibility of that the entire available channels are not able to support the successful packet transmission based on spectrum handoff, the probability of spectrum handoff is equal to:

$$P_{sh} = \sum_{i=1}^{L} \pi(T_i)(p_{uspt,c_i} - \prod_{j=1}^{L} p_{uspt,c_j}) \tag{13}$$

In such a case, the troubled nodes do not perform spectrum handoff. These nodes have two choices. In such a case, the troubled nodes do not perform spectrum handoff; they perform local flow handoff. In (13), the term $\sum_{i=1}^{L} \pi(T_i) \prod_{j=1}^{L} p_{uspt,c_j}$ is the probability that the entire available channel cannot support the successful packet transmission. This term can be defined as the probability of local rerouting:

$$P_{lh} = \sum_{i=1}^{L} \pi(T_i) \prod_{j=1}^{L} p_{uspt,c_j} \tag{14}$$

The probability of successful spectrum handoff (P_{ssh}) depends on the probability of successful packet transmission in a hop $p_{spt,c}$. However, to perform local flow handoff, equations (11) and (12) must be satisfied.

We define the link maintenance probability (P_{LM}) as the probability that the link is successfully maintained during unsuccessful packet transmission in a hop or between troubled nodes, which is dependent on the probability of channel availability between two nodes. Thus, the link maintenance probability, considering only spectrum handoff, can be written as follows:

$$P_{LM,sh} = P_{sh} P_{spt,c} \tag{15}$$

When the link maintenance is not successful, despite the spectrum handoff, local rerouting is performed. In this case, the probability of link maintenance is as follows:

$$P_{LM,lh} = \left(\frac{N-2}{N}\right)(1 - P_{LM,sh}) P_{lh} P_{spt,c} \tag{16}$$

Finally, the probability of link maintenance considering the integrated routing and spectrum handoff management can be written as follows:

$$P_{LM,Ih} = P_{LM,sh} + P_{LM,lh} \tag{17}$$

6 Results and Discussion

Spectrum heterogeneity and SU mobility have a significant effect on the handoff blocking probability in the integrated routing and spectrum handoff management

scheme. We define the probability of unsuccessful link maintenance as the probability of spectrum handoff blocking (P_{hb}). Fig. 6 compares the probability of unsuccessful rerouting (P_{usrr}), the handoff blocking probability deploying only spectrum handoff ($P_{hb,sh}$) and the handoff blocking probability deploying integrated routing and spectrum handoff management ($P_{hb,Ih}$), considering various number of SU nodes in the network. In the following figures, L=2, p=0.5, R_1=75 m, R_2=125 m, R_T=150 m, C=10, c_1=5, c_2=5, and the number of hops is equal to 14. Based on this figure, the integrated routing and spectrum handoff management scheme outperforms the scheme only deploying spectrum handoff in terms of link maintenance probability and spectrum handoff blocking probability. The probability of link maintenance in the integrated routing and spectrum handoff scheme is also significantly higher than the probability of successful rerouting. As the number of SUs in the network increases, the probability of handoff blocking decreases because the probability of finding the proper nodes to perform local rerouting increases.

Based on [9], with an area network of A_N=1000 m^2 and R_T=150 m, the number of expected hop count in the network is equal to 5. Fig. 7 compares the proposed parameters in Fig .6 with an expected hop count in the network equal to five. Comparing Fig. 6 and Fig. 7, we conclude that the probability of handoff blocking decreases when the number of hops decreases. This results shows that the proposed integrated spectrum handoff management and routing scheme achieves more actual data transmission opportunities.

Fig. 8 compares the effect of channel heterogeneity and channel homogeneity on handoff blocking probability *(P_{hb})*. This figure implies that, in contrast to homogenous channel condition, the node density in the network with the heterogeneous channel must be high to have an acceptable P_{hb}. Hence, we must consider channel heterogeneity in terms of transmission range and path loss.

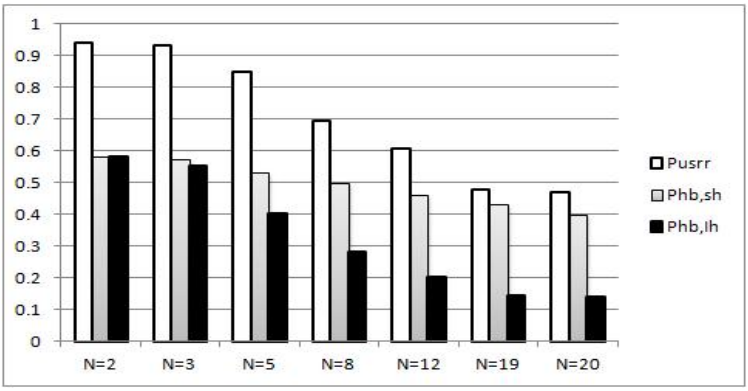


Fig. 6. Comparison of different handoff management schemes and routing performance with a hop count of 14

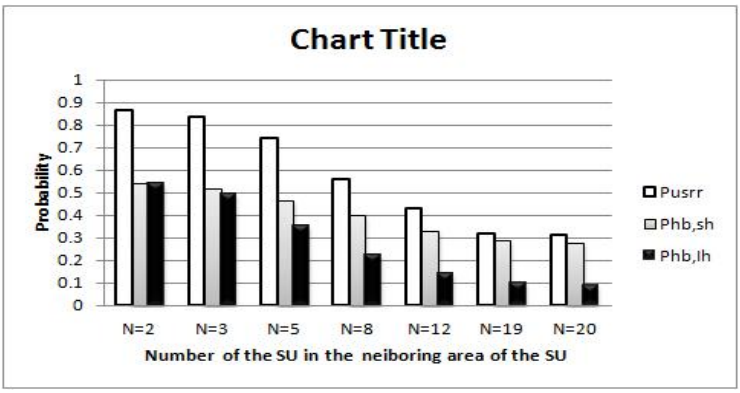


Fig. 7. Comparison of different handoff management schemes and routing performance with an expected hop count of 5

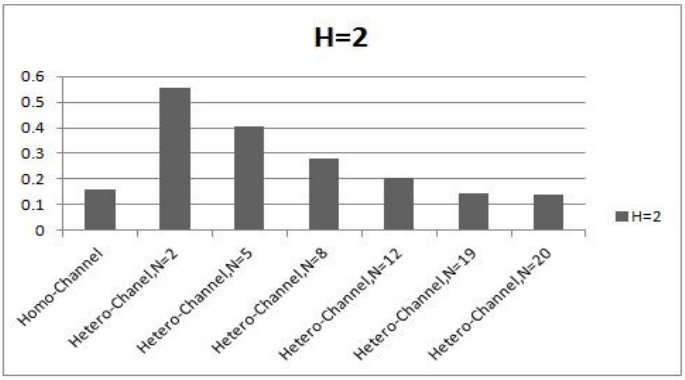


Fig. 8. Effect of channel heterogeneity on integrated handoff blocking probability

7 Conclusion

Spectrum handoff management is still an open issue in CR networks. It is particularly challenging in CR-MANETs. In CR-MANETs, the available spectrum bands vary over time and space, while they are distributed non-adjacently over a broad frequency range. However, in CR-MANETs, the fluctuation of PU activity and the SU mobility make the issue of maintaining optimal routes more complex. In this work, we present an integrated spectrum handoff management and routing scheme that considers spectrum mobility in the time and space domains and considers the network topology variations. We propose a network architecture that considers the heterogeneous spectrum availability and its variation over time and space and distributed nodes. Then, we formalise the probability of channel availability in this dynamic radio environment. Based on this unified architecture, an integrated routing and spectrum handoff management scheme is proposed. The proposed scheme considers the CR-MANETs

spectrum handoff problem and incorporates the routing issue. It is a step towards a comprehensive management system for CR-MANETs.

Acknowledgements. The authors would like to thank all those who contributed toward making this research successful. Also, we would like to thank all reviews for their insightful comments. The authors wish to express their gratitude to Ministry of Higher Education (MOHE), Malaysia and Research Management Center (RMC), Universiti Teknologi Malaysia for the financial support of this project under GUP research grant no: Q.J130000.2523.04H89.

References

1. Min, A.W., Shin, K.G.: Impact of mobility on spectrum sensing in cognitive radio networks. In: Proceedings of the 2009 ACM Workshop on Cognitive Radio Networks, CoRoNet 2009, p. 13 (2009)
2. Giupponi, L., Perez-Neira, A.I.: Fuzzy-based Spectrum Handoff in Cognitive Radio Networks. In: 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications CrownCom, pp. 1–6. IEEE Press (2008)
3. Duan, J., Li, Y.: An optimal spectrum handoff scheme for cognitive radio mobile Ad hoc networks. AECE, 11–16 (2011)
4. Song, Y., Xie, J.: ProSpect: A Proactive Spectrum Handoff Framework for Cognitive Radio Ad hoc Networks without Common Control Channel. IEEE Transactions on Mobile Computing 3(99), 1 (2011)
5. Song, Y., Xie, J.: Performance Analysis of Spectrum Handoff for Cognitive Radio Ad hoc Networks without Common Control Channel under Homogeneous Primary Traffic. Compare a Journal of Comparative Education, 3011–3019 (2011)
6. Samad, N., Sharifah Kamilah, S.Y., Nurul Muazzah, A.L., Vahid, A.: Proactive Integrated handoff management in CR-MANETs: A conceptual model. In: 2012 IEEE Symposium on Wireless Technology and Applications (ISWTA), pp. 33–38 (2012)
7. Samad, N., Josh, A.A., Sharifah Kamilah, S.Y., Nurul Muazzah, A.L., Vahid, A.: Characterization of spectrum mobility and channel availability in CR-MANETs. In: Proc. of the International Conference on Advances in Mobile Networks and Communication, MNC 2012, pp. 12–16 (2012)
8. Abhilash, P., Perur, S., Iyer, S.: Router Handoff: An Approach for Preemptive Route Repair in Mobile Ad Hoc Networks. In: Sahni, S.K., Prasanna, V.K., Shukla, U. (eds.) HiPC 2002. LNCS, vol. 2552, pp. 347–357. Springer, Heidelberg (2002)
9. Li, J., Blake, C., De Couto, D.S.J., Lee, H.I., Morris, R.: Capacity of Ad Hoc wireless networks. Network 01, 61–69 (2001)

Dimensioning Self-sufficient Networks of Energy Harvesting Embedded Devices

Nicola Bui[1,2,⋆] and Michele Rossi[2,3]

[1] Patavina Technologies, 59/8 via Venezia, 35131, Padova, Italy
nicola.bui@patavinatech.com
[2] DEI, University of Padova, 6/b via Gradenigo, 35131, Padova, Italy
michele.rossi@dei.unipd.it
[3] Consorzio Ferrara Ricerche, 1 via Saragat, 44124, Ferrara, Italy⋆⋆

Abstract. Energy efficiency and self sustainability are among the primary objectives for networks of embedded devices, such as those of sensor networks and Internet of Things. In this paper we present a reference framework to obtain the optimal configuration parameters of networked devices with energy scavenging capabilities. Specifically, we derive an optimization method that links a simple and yet effective energy consumption model to network topology configurations and to the average energy that is harvested from the environment. This model is efficiently solved using interior point algorithms, making it possible to obtain optimal communication parameters and their feasibility regions, so as to ensure the perpetual operation of embedded communicating devices. Moreover, our framework allows for a dynamic system configuration as a function of the harvested energy income rate, thus making the considered networks flexible and self-adaptable.

Keywords: Sensor Networks, Energy Harvesting, System Design, Optimization, Embedded, Networking, Communication.

1 Introduction

There is a sufficiency in the world for man's need but not for man's greed (Mohandas K. Gandhi) – as expressed by Gandhi, the world has enough resources to satisfy everyone needs, but only if these are consumed wisely. Similarly, when addressing the problem of energy self-sustainability of constrained networked devices, the energy consumed must be carefully matched to that harvested from the environment. In the literature, several works have been proposed towards this end. Examples are [10], for a comprehensive state of the art analysis on energy harvesting devices for embedded systems, [8,9], for the analysis of optimal transmission and routing policies and [7] for the characterization of solar

⋆ Corresponding author.

⋆⋆ The research leading to these results has received funding from the Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 251557 (Project SWAP).

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 138–150, 2013.

powered sensor networks. However, we observe that most of the work in the literature either focuses on experimental systems, discussing their implementation and achievable performance, or on theoretical models, that are often rather involved and whose results are hardly applicable to practical network protocols. In this paper, we aim at filling this gap by presenting a tractable model that allows the dimensioning and adaptation of the key parameters of widely adopted networking protocols. For the channel access we adopt the Low Power Listening (LPL) MAC [1,3], whereas routing dynamics are modeled through the IETF Routing for low Power Lossy networks (RPL) [4,6].

In short, the main contributions of this paper are:

1. a simple but accurate model for the energy consumption of a network of embedded devices, in Section 2.1;
2. an original network topology model able to link communication dynamics to network structures, in Section 2.2;
3. a framework to dimension the communication protocol parameters so as to allow the perpetual operation of the network, in Section 2.3.

Note that our framework allows for a dynamic system adaptation based on the actual energy income rate, thus making embedded networked devices self-sufficient and flexible to energy fluctuations.

2 System Analysis

2.1 Device Behavior

The energy consumption of embedded multi-hop networks can be split in two parts: E1) the energy associated with communication activities (due to the microprocessor and the radio transceiver), and E2) the energy related to other operations, such as data sampling and processing (due to the microprocessor and its peripherals).

Moreover, while E2 can be assumed to be equal for all network devices, E1 depends on their actual position in the data gathering tree. In fact, for data retrieval applications, the devices that are placed closer to the sink, in addition to their own data (*endogenous* traffic), also have to forward data for the devices that are placed farther away (*exogenous* traffic). This leads to a progressive increase in the traffic load as we move closer to the sink. In what follows, we refer to the devices that are directly connected to the sink as *bottleneck devices*, as these will have to carry the highest amount of data and, in turn, are subject to the highest energy expenditure. Hence, bottleneck devices are critically relevant for the lifetime of a network and in our following analysis we focus on the design of their transmission policies, so as to provide worst-case protocol rules that ensure the self-sustainability of the entire system.

Energy States. The average energy consumption is now specified for each of the most relevant operational states of an embedded device. The energy expenditure

is expressed in terms of average amount of current drained per activity, as this will facilitate our subsequent calculations related to the battery capacity.

TX: The energy associated with data transmission, E_{TX}, depends on the total current drawn, obtained as the sum of the current drawn by the microprocessor, I_{CPU}, plus that drawn by the transceiver in its transmitting state, I_{TX}, and the transmission rate, r_{TX}, defined as the average fraction of time spent in the TX state. On average, $r_{\mathrm{TX}} = f_{\mathrm{TX}} t_{\mathrm{TX}}$, where t_{TX} is the time required by the transmit operation and f_{TX} is its frequency. Hence, $E_{\mathrm{TX}} = (I_{\mathrm{CPU}} + I_{\mathrm{TX}}) r_{\mathrm{TX}}$.

RX: The energy associated with message reception, $E_{\mathrm{RX}} = (I_{\mathrm{CPU}} + I_{\mathrm{RX}}) r_{\mathrm{RX}}$, with $r_{\mathrm{RX}} = f_{\mathrm{RX}} t_{\mathrm{RX}}$, where r_{RX}, f_{RX} and t_{RX} are the average fraction of time spent in RX, the average number of packets received per second and the packet reception time, respectively.

INT: The energy associated with the reception of messages destined to other nodes is referred to as *interference*, and $E_{\mathrm{INT}} = (I_{\mathrm{CPU}} + I_{\mathrm{RX}}) r_{\mathrm{INT}}$, because the current drawn by the transceiver due to interference is the same of that drawn during regular receptions. $r_{\mathrm{INT}} = f_{\mathrm{INT}} t_{\mathrm{INT}}$, where f_{INT} and t_{INT} represent the number of interfering packets received per second and the time spent by the transceiver for each of them, respectively.

CPU: The microprocessor activity not related to communication such as data sampling, data storage into the internal flash memory, etc. In this case, the energy only depends on the current drawn by the microprocessor, I_{CPU}. Hence, $E_{\mathrm{CPU}} = I_{\mathrm{CPU}} r_{\mathrm{CPU}}$, where $r_{\mathrm{CPU}} = f_{\mathrm{CPU}} t_{\mathrm{CPU}}$.

IDLE: A device is in IDLE whenever it is not involved in any of the previous activities; thus, $r_{\mathrm{IDLE}} = 1 - \sum_x r_x, x \in S$, where $S = \{\mathrm{TX}, \mathrm{RX}, \mathrm{INT}, \mathrm{CPU}\}$ is the set of all the previous operational modes. The IDLE state includes two sub-states, depending on whether the transceiver is on or off. Specifically:

- **Comm OFF:** In this state the device is sleeping, the microprocessor and the transceiver are into their most energy efficient power modes and the total current drawn is I_{IDLE}. Thus, we have $E_{\mathrm{IDLE-OFF}} = I_{\mathrm{IDLE}} r_{\mathrm{IDLE}} r_{\mathrm{OFF}}$, where r_{OFF} is the fraction of time during which the transceiver is off and the device is in IDLE (i.e., it is not involved in any of the activities in S).
- **Comm ON:** In this state the LPL MAC performs clear channel assessment (CCA) in order to detect incoming transmissions. Thus, the transceiver is on and the current drawn during CCA equals that drawn in RX, thus: $E_{\mathrm{IDLE-ON}} = (I_{\mathrm{CPU}} + I_{\mathrm{RX}}) r_{\mathrm{IDLE}} r_{\mathrm{ON}}$, where $r_{\mathrm{ON}} = 1 - r_{\mathrm{OFF}}$.

Concluding, the total average current consumed by an embedded device, E_{tot}, is expressed as:

$$E_{\mathrm{tot}} = E_{\mathrm{TX}} + E_{\mathrm{RX}} + E_{\mathrm{INT}} + E_{\mathrm{CPU}} + E_{\mathrm{IDLE-ON}} + E_{\mathrm{IDLE-OFF}} \,. \qquad (1)$$

In the following, we specialize the contributions in (1) in order to account for the MAC protocol behavior, the data collection algorithm, and the related networking functionalities.

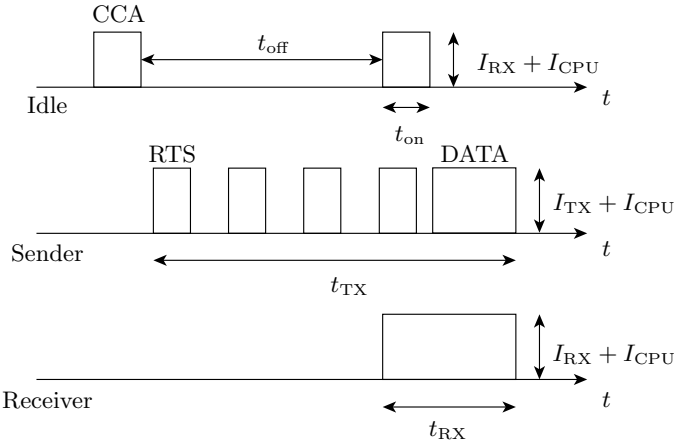


Fig. 1. MAC timings for CCA (top), TX (middle) and RX phases (bottom)

The MAC Contribution. usually depends on the radio duty cycle, which, in turn, depends on the duration of the active and the sleep phases, t_{on} and t_{off}, respectively. Devices that implement LPL (or other preamble-sampling MAC protocols) periodically sample the channel, during the CCA phase, to detect incoming transmissions. Hence, we have $r_{\text{ON}} = t_{\text{on}}/(t_{\text{on}} + t_{\text{off}})$, and $r_{\text{OFF}} = t_{\text{off}}/(t_{\text{on}} + t_{\text{off}})$ as the devices are active for t_{on} seconds in every cycle. Fig. 1 provides a graphical representation of the CCA phase (top figure), the data transmission phase (middle), and the data reception phase (bottom).

For what concerns the transmission operation, in order for a device to be sure to wake up its intended receiver, a number of request to send (RTS) packets has to be sent before transmitting the data packet. This phase is referred to as *channel probing* and lasts at most $t_{\text{on}} + t_{\text{off}}$ seconds. Once the receiver detects an RTS (of duration t_{RTS}), it sends back to the transmitter a clear to send (CTS) message (t_{CTS}), which is followed by the transmission of the DATA packet (t_{DATA}) and the corresponding acknowledgment (ACK, of duration t_{ACK}). In the rest of the paper, to simplify the notation we indicate the sum of CTS, ACK and DATA times as, t_{DATA}, and we assume the current drawn by the device during the entire TX phase as constant and equal to $I_{\text{CPU}} + I_{\text{TX}}$. Hence, we can write $t_{\text{TX}} \leq t_{\text{on}} + t_{\text{off}} + t_{\text{DATA}}$. In the analysis that follows, we consider the equality sign in t_{TX} as this leads to the worst case in terms of energy expenditure.

The reception phase does not depend on the duty cycle, but only on the specific radio transceiver in use. In fact, the receive time consists of the sum of the time taken to receive an RTS, to receive the DATA and to transmit the corresponding ACK. To simplify the notation, we assume $t_{\text{RTS}} = t_{\text{CTS}}$ so that we can still use t_{DATA} to represent the time spent to receive a data packet.

Finally, the time spent in state RX when overhearing interfering transmissions only depends on the time needed to receive and decode one RTS packet. This is because it is sufficient to read the destination address in the header of the

RTS to promptly abort the RX procedure and bring the device back into IDLE. Thus, $t_{\mathrm{INT}} < t_{\mathrm{DATA}}$ and t_{INT} is still independent of the duty cycle.

The Data Collection. application influences the frequencies f_{TX}, f_{RX} and f_{INT}. The analysis that follows focuses on the *bottleneck* devices, that are placed within the first hop of the data collection tree. In fact, these are the nodes that determine the network lifetime. In what follows, we refer to the children nodes of a node located within hop $i \geq 1$ as n_{Ci}.

Most data collection applications depend on a given information update time, t_{U}, which dictates the timings for sensor sampling and information transmission. Note that, our model still holds on average for applications using adaptive timings by considering their average update time for t_{U}. Moreover, a given bottleneck node has to relay the data traffic generated by its children nodes, where n_{C1} is their number. Given this, a single bottleneck device will send endogenous data to the sink node every t_{U} seconds, and will relay exogenous data for its children nodes every t_{U}/n_{C1} seconds (assuming that all nodes in the network are configured with the same t_{U}). Thus, its reception and transmission frequencies are $f_{\mathrm{RX,DC}} = n_{C1}/t_{\mathrm{U}}$ and $f_{\mathrm{TX,DC}} = (1 + n_{C1})/t_{\mathrm{U}}$, respectively.

To determine the frequency of interfering transmissions we need a further element: the number of interfering nodes, n_I. In our analysis, we assume that a bottleneck node receives interference from the devices in the first three hops. With n_{I1}, n_{I2} and n_{I3} we represent the number of interfering nodes in the first, second and third hop from the sink, respectively, and $n_I = n_{I1} + n_{I2} + n_{I3}$. Thus, a bottleneck node is subject to $f_{\mathrm{INT,DC}} = n_{\mathrm{INT}}/t_{\mathrm{U}}$ interfering transmissions per second, where $n_{\mathrm{INT}} = (n_{I1}(n_{C1} + 1) + n_{I2}(n_{C2} + 1) + n_{I3}(n_{C3} + 1))$.

Finally, f_{CPU} is inversely proportional to t_{U}, according to the time needed to sample the data to be sent within a packet. For instance, if 10 sensor readings are to be stored in a packet, $f_{CPU} = 10/t_{\mathrm{U}}$. Thus, the frequencies associated with the data gathering activity can be written as:

$$\begin{aligned} f_{\mathrm{TX,DC}} &= (1 + n_{C1})/t_{\mathrm{U}} \\ f_{\mathrm{RX,DC}} &= n_{C1}/t_{\mathrm{U}} \\ f_{\mathrm{INT,DC}} &= n_{\mathrm{INT}}/t_{\mathrm{U}} \\ f_{\mathrm{CPU,DC}} &= K_{\mathrm{U}}/t_{\mathrm{U}} \ , \end{aligned} \tag{2}$$

where K_{U} is a constant linking the CPU activity to the data collection rate.

The Network Traffic. is modeled similarly to what done for the data collection application. In particular, since we adopt RPL as the routing mechanism, every node is subject to additional control traffic, needed for the creation and maintenance of the routes towards the sink. Although during the network setup phase and after each path recovery RPL increases its control message rate, our analysis focuses on RPL's steady state behavior, which starts when the trickle timer reaches its maximum update time, t_{RPL}.

In the steady state, RPL control messaging consists of: i) the DIO dissemination, originated by the sink every t_{RPL} and directed to all nodes; ii) the DAO confirmation, originated by every node and directed towards the sink.

For what concerns the bottleneck node, its networking frequencies are:

$$\begin{aligned} f_{\text{TX,RPL}} &= f_{\text{TX,DIO}} + f_{\text{TX,DAO}} = 1/t_{\text{RPL}} + (1 + n_{C1})/t_{\text{RPL}} \\ f_{\text{RX,RPL}} &= f_{\text{RX,DIO}} + f_{\text{RX,DAO}} = 1/t_{\text{RPL}} + n_I/t_{\text{RPL}} + n_{C1}/t_{\text{RPL}} \\ f_{\text{INT,RPL}} &= f_{\text{INT,DAO}} = n_{\text{INT}}/t_{\text{RPL}} \ , \end{aligned} \tag{3}$$

where the interference equation is missing the DIO term because DIO messages are sent using broadcast addresses, hence each node must receive every DIO.

Finally, summing (2) and (3), and taking the expectation where needed, we obtain:

$$\begin{aligned} f_{\text{TX}} &= f_{\text{TX,DC}} + f_{\text{TX,RPL}} = (1 + n_{C1})/t_{\text{U}} + (2 + n_{C1})/t_{\text{RPL}} \\ f_{\text{RX}} &= f_{\text{RX,DC}} + f_{\text{RX,RPL}} = n_{C1}/t_{\text{U}} + (1 + n_I + n_{C1})/t_{\text{RPL}} \\ f_{\text{INT}} &= f_{\text{INT,DC}} + f_{\text{INT,RPL}} = n_{\text{INT}}/t_{\text{U}} + n_{\text{INT}}/t_{\text{RPL}} \\ f_{\text{CPU}} &= f_{\text{CPU,DC}} = K_{\text{U}}/t_{\text{U}} \ . \end{aligned} \tag{4}$$

Hence, (1) together with the frequencies in (4) return the system energy consumption in terms of average current drained, subject to $r_x > 0, \forall\, x \in S$ and to the following rate constraint:

$$\sum_{x \in S} r_x < 1 \ . \tag{5}$$

2.2 Network Topology

Next, we present a convenient model to build data gathering trees with properties resembling those of actual sensor network deployments. These topologies will be subsequently used to derive the parameters n_{Ii} and n_{Ci}, for $i = 1, 2, 3$, which will drive the overall optimization of Section 2.3.

Our objective is that of building data collection trees by being able to control certain topological properties such as their *hop count depth* $H \geq 1$ and their *expansion factor*, $\xi \in (0, +\infty)$, which corresponds to the ratio between the number of nodes in a generic level i of the tree and the number of nodes that are directly connected to them and belong to the preceding level $i - 1$. For the sake of simplicity of illustration and without loss of generality, in what follows we consider the case where ξ is a positive integer.

Construction of the connectivity graph. All the sensor nodes are arranged according to a tree topology having a data collector device (the sink) as the root of the tree (level 0) and $H \geq 1$ levels. The root is directly connected to a number of children nodes n_1 which belong to the first level of the tree. Thus, a further number $n_2 = \xi n_1$ of sensors is placed in the second level of the tree so that each of the n_1 nodes in the preceding level 1 is directly connected to ξ nodes in the next level 2. This construction process is iterated for each subsequent level until we connect the $n_H = \xi n_{H-1}$ nodes belonging to the last level H. Again, each of

the nodes in level $H-1$ is connected to exactly ξ nodes in the last level H. ξ is the expansion factor of the so obtained network topology. Note that the total number of nodes in the tree, excluding the root, is found as:

$$N = \sum_{i=1}^{H} n_i = \sum_{i=1}^{H} \xi^{i-1} n_1 = \begin{cases} Hn_1 & \xi = 1 \\ \dfrac{1-\xi^H}{1-\xi} n_1 & \text{elsewhere} \end{cases}. \tag{6}$$

We remark that the nodes belonging to the first level of the tree are the most loaded and also that their lifetime corresponds to the lifetime of the entire network as, whenever their batteries run out of charge, the entire network gets disconnected. The focus of this paper is on transmission policies assuring the perpetual operation of the network when node batteries are replenished thanks to the energy harvested from the environment. The nodes in the first level of the tree deserve our special attention to achieve this goal, as the optimal policies in terms of network lifetime maximization depend on their status in terms of: i) total traffic that they have to serve and ii) the level of interference that they experience. Regarding i), we note that each of the n_1 nodes residing in the first level of the tree receives upstream (from the nodes to the root) traffic from $n_{C1} = \sum_{i=1}^{H-1} \xi^i$ nodes, similarly we have $n_{C2} = \sum_{i=1}^{H-2} \xi^i$ and $n_{C3} = \sum_{i=1}^{H-3} \xi^i$. Point ii) is characterized next.

Approximation of the number of interfering nodes. In what follows we compute the number of nodes that, on average, interfere with the nodes in the first level of the tree. To that extent, we need to specify a model covering the geographical deployment of the nodes along with their transmission and interference ranges. For the placement of the nodes, we consider concentric circular areas, whose radii vary in steps of R units, where R is also the transmission range of the nodes. The sink is directly connected to the first n_1 children nodes, that are placed uniformly at random in a first circular area $\mathcal{A}_1$ of radius R, which has the root as its center. Thus, the next n_2 of sensors are placed uniformly at random within the ring (referred to as $\mathcal{A}_2$) surrounding $\mathcal{A}_1$ and having R and $2R$ as its inner and outer radii, respectively. The process continues for each subsequent external ring: the generic ring i ($\mathcal{A}_i$), for instance, extends from $(i-1)R$ to iR and contains ξn_{i-1} nodes, placed again uniformly at random in $\mathcal{A}_i$. We continue as explained until we place the n_H nodes of the last level H in the in the ring $\mathcal{A}_H$, which extends from $(H-1)R$ to HR. In addition, we consider that the interfering range is $R_I = \alpha R$, with $\alpha \in [1, 2]$. Note that a random and uniform distribution of the nodes is an approximation and does not exactly match the geographical distribution that arises from the considered connectivity graph. Nevertheless, we deem it reasonable and it allows for a tractable analysis.

To calculate the average number of interferers for any given node in the first level of the tree, consider the diagram of Fig. 2. For the analysis, we focus on a tagged node d in the first level of the tree ($\mathcal{A}_1$ of radius R) with distance $x < R$ from the root, representing its interference area $\mathcal{A}_I$ through a dashed circle of radius αR, considering for this example $\alpha > 1$. Since the expansion

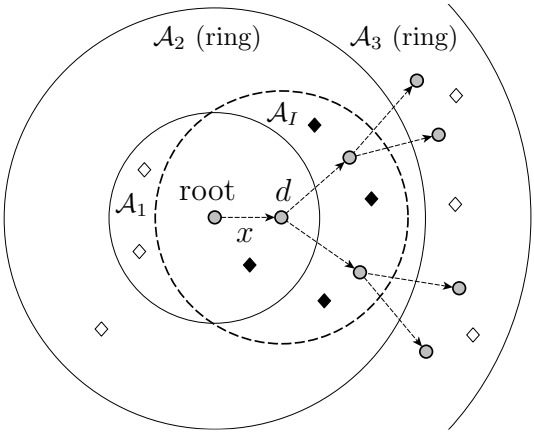


Fig. 2. Network with expansion factor $\xi = 2$ and $H = 3$. Node d receives interference from the nodes that are not connected with it (black-filled diamonds in the figure) but that are within its interference range (dashed circle $\mathcal{A}_I$).

factor is $\xi = 2$, this node has two children nodes in the next level ($\mathcal{A}_2$) and each of these has two further children in the last level 3. Of course, the root and the children nodes are not interferers for node d. However, the nodes indicated with black-filled diamonds in the figure, that may be located either in the intersection $\mathcal{A}_{1\cap I} = \mathcal{A}_1 \cap \mathcal{A}_I$ or in $\mathcal{A}_{2\cap I} = \mathcal{A}_2 \cap \mathcal{A}_I$ do interfere with node d's transmission activity. Given this, finding the total number of interferers n_I for node d amounts to finding the sum of the interfering nodes in $\mathcal{A}_{1\cap I}$, denoted by x_{I1}, and those in $\mathcal{A}_{2\cap I}$, denoted by n_{I2}, i.e., $n_I = n_{I1} + n_{I2}$. Note that n_{I1} and n_{I2} are random variables depending on the area of the respective intersections.

We now obtain $E[n_I]$ for the general case where $\alpha \in [1, 2]$. To this end, note that when $x > (2 - \alpha)R$, potential interferers for node d are located in $\mathcal{A}_{1\cap I}$, $\mathcal{A}_{2\cap I}$ and also $\mathcal{A}_{3\cap I} = \mathcal{A}_3 \cap \mathcal{A}_I$ and we have $n_I = n_{I1} + n_{I2} + n_{I3}$, where n_{I3} is the number of interferers in $\mathcal{A}_{3\cap I}$. We also note that $f(x) = 2x/R^2$ is the pdf of the distance x between node d and the root of the tree. After some algebra, we can write $E[n_I] = E[n_{I1}] + E[n_{I2}] + E[n_{I3}]$, with:

$$E[n_{Ii}] = \int_0^R f(x)(n_i - \xi^{i-1})p_i(x)\mathrm{d}x,\ i = 1, 2;\ E[n_{I3}] = \int_{(2-\alpha)R}^R f(x)n_3 p_3(x)\mathrm{d}x\ , \tag{7}$$

where $p_k(x)$ is the probability that a node will be in the intersection $\mathcal{A}_{k\cap I}$, given that it is placed uniformly at random in $\mathcal{A}_k$ and is obtained as:

$$p_1(x) = \frac{M_{1\cap I}(x)}{\pi R^2}\ ,\ p_2(x) = \frac{M_{2\cap I}(x)}{3\pi R^2}\ ,\ p_3(x) = \frac{M_{3\cap I}(x)}{5\pi R^2}\ , \tag{8}$$

where we refer to $M_{k\cap I}(x)$ as the area of the intersection $\mathcal{A}_{k\cap I}$, with $k = 1, 2, 3$, which is obtained through standard formulas for the intersection of circles [5].

2.3 Optimal Transmission Policies

We consider a renewable energy source that can be in any of the two states H and L. When the source is in state H, the sensor devices harvest an average current I_{H} (obtained after the input regulator [7]) and the source remains in state H for t_{H} seconds, where t_{H} is a random variable with mean $\overline{t}_{\mathrm{H}} = E[t_{\mathrm{H}}]$. After that, the source moves to state L, where the harvested current is $I_{\mathrm{L}} \ll I_{\mathrm{H}}$ and the source remains in this state for t_{L} seconds, where $\overline{t}_{\mathrm{L}} = E[t_{\mathrm{L}}]$, after which it moves back to state H. This can model the energy that is typically harvested by a solar panel during daytime (state H) and the night (L). Given this, the quantity of energy harvested during an entire cycle of duration $t_{\mathrm{H}} + t_{\mathrm{L}}$ is: $\overline{E}_{\mathrm{harv}} = I_{\mathrm{H}}\overline{t}_{\mathrm{H}} + I_{\mathrm{L}}\overline{t}_{\mathrm{L}}$.

On the other hand, the average amount of energy that is drained by a bottleneck device during an entire source cycle is: $\overline{E}_{\mathrm{out}}(t_{\mathrm{U}}, t_{\mathrm{off}}) = \overline{E}_{\mathrm{tot}}(t_{\mathrm{U}}, t_{\mathrm{off}})(\overline{t}_{\mathrm{H}}+\overline{t}_{\mathrm{L}})$, where we have made the dependence on t_{U} and t_{off} explicit, as these are the parameters that we want to optimize in our design. Also, $\overline{E}_{\mathrm{tot}}(t_{\mathrm{U}}, t_{\mathrm{off}})$ is defined as the expectation of E_{tot} in (1) for given topological parameters n_1, ξ and H.[1]

Condition for self-sufficiency. Our optimization approach corresponds to deciding the operational state of the devices, that in this paper amounts to setting their variables t_{U} (transmission rate) and t_{off} (duty cycle), whenever the source enters state H. Thus, the policy is used for an entire source cycle and adapted again at the beginning of the next one. Given this, the condition that has to be satisfied to ensure the self-sufficiency of the bottleneck devices, i.e., that they will be able to live unattended by carrying out their activities, while fully recharging their battery during an entire renewable source cycle, corresponds to assuring that the energy consumed by the device during an entire source cycle ($\overline{E}_{\mathrm{out}}(t_{\mathrm{U}}, t_{\mathrm{off}})$) is no larger than the energy inflow during the same time interval ($\overline{E}_{\mathrm{harv}}$). Hence, we get the following constraint:

$$\overline{E}_{\mathrm{tot}}(t_{\mathrm{U}}, t_{\mathrm{off}}) \leq \frac{I_{\mathrm{H}}\overline{t}_{\mathrm{H}} + I_{\mathrm{L}}\overline{t}_{\mathrm{L}}}{\overline{t}_{\mathrm{H}} + \overline{t}_{\mathrm{L}}} \stackrel{\mathrm{def}}{=} K_{\mathrm{H,L}} \,, \tag{9}$$

where $K_{\mathrm{H,L}}$ is a constant that depends on the energy source dynamics.

Optimization problem. Given the above definitions and condition pertaining to the self-sustainability, we now write an optimization problem to find the value of t_{U} and t_{off} that maximize the throughput $1/t_{\mathrm{U}}$ while respecting the rate feasibility constraint (5) and the self-sustainability constraint (9):

$$\begin{aligned} \underset{t_{\mathrm{U}}, t_{\mathrm{off}}}{\text{minimize}} \quad & t_{\mathrm{U}} \\ \text{subject to} \quad & \overline{E}_{\mathrm{tot}}(t_{\mathrm{U}}, t_{\mathrm{off}}) \leq K_{\mathrm{H,L}} \\ & \sum_{x \in S} r_x(t_{\mathrm{U}}, t_{\mathrm{off}}) < 1 \\ & t_{\mathrm{U}} > 0 \,,\; t_{\mathrm{off}} > 0 \,. \end{aligned} \tag{10}$$

[1] Note that this expectation is needed due to the random placement of the interfering nodes in the first three hops.

Here, we have again emphasized the dependence of the rates on the optimization variables t_{U} and t_{off}. Inspection of the constraints shows that this is a non-linear optimization problem. Through some algebraic manipulations, (10) can be rewritten as a geometric program, which is solvable through highly efficient interior point algorithms [2].

3 Performance Evaluation

Our dimensioning analysis focuses along two directions: the impact of the available energy income and the effect of the network topology. For the energy source, we refer to an harvester based on a solar cell, referring to L and H and daytime and night, respectively. In the following we discuss results about two scenarios:

S1: the topology parameters are fixed to 5 nodes in the first hop ($n_1 = 5$), 5 hops ($H = 5$), $\xi = 1.5$, and the interference range is 1.5 times the transmission range ($\alpha = 1.5$). In this scenario, we fixed $I_{\mathrm{L}} = 0$ (e.g., no energy during the night) and we let I_{H} and $\overline{t}_{\mathrm{H}}/(\overline{t}_{\mathrm{H}}+\overline{t}_{\mathrm{L}})$ vary between 0.1 and 10000 mA and between 0.1 and 1, respectively.

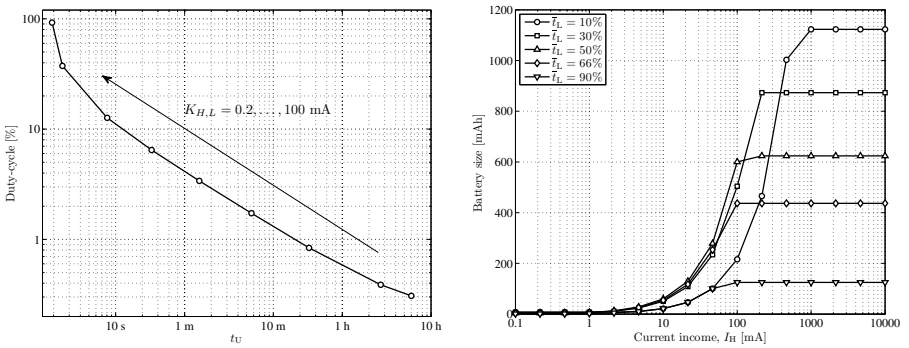
S2: here we fixed $I_{\mathrm{H}} = 10$ and $I_{\mathrm{L}} = 0$ mA and we let the network size vary in terms of first hop nodes, $n_1 \in [1, \ldots, 5]$, max hop count, $H \in [1, \ldots, 5]$, expansion factor, $\xi \in [1, 2]$ for a constant interference range parameter $\alpha = 1.5$. In addition, in order to account for the varying amounts of sunlight received during the year, we considered $\overline{t}_{\mathrm{H}}/(\overline{t}_{\mathrm{H}} + \overline{t}_{\mathrm{L}}) = 0.6$ for summer time and $\overline{t}_{\mathrm{H}}/(\overline{t}_{\mathrm{H}} + \overline{t}_{\mathrm{L}}) = 0.3$ for winter. The system parameters are given in Table 1.

Table 1. System parameters

t_{on}	t_{DATA}	t_{INT}	t_{CPU}	t_{RPL}	I_{TX}	I_{RX}	I_{CPU}	I_{IDLE}
6 ms	14 ms	10 ms	40 ms	6 h	14 mA	12.3 mA	42 mA	31 μA

For S1, we obtain a network having a total of 68 nodes, where the bottleneck node has $n_{C1} = 15$ children nodes and $n_I = 8$ ($n_{I1} = 4, n_{I2} = 2$ and $n_{I3} = 2$) interfering nodes. Results obtained for S1 are shown in Fig. 3, where Fig. 3(a) shows the optimal values for t_{U} and t_{off} varying $K_{\mathrm{H,L}}$ from 0.2 to 100 mA and Fig. 3(b) shows the *optimal battery size*. For a certain $K_{\mathrm{H,L}}$, the optimal battery size is the one that guarantees that the node remains operational, using the optimal t_{U} and t_{off}, for the entire duration of state L. Hence, the expected optimal battery size is $\overline{E}_{\mathrm{tot}}(t_{\mathrm{U}}, t_{\mathrm{off}})\overline{t}_{\mathrm{L}}$.

From these figures we conclude the following. As expected, increasing the average available energy harvested, $K_{\mathrm{H,L}}$, leads to shorter update times, t_{U}, and higher duty cycles (shorter t_{off}) down to $t_{\mathrm{U}} = 2$ s with nodes duty cycle close to 100 %. The least current needed to make the network operational is 0.2 mA with duty cycle 0.3 % and sending endogenous data every 6 hours (leftmost point of Fig. 3(a)). Increasing the current above 100 mA does not provide any improvement, as the system is limited by (5), while decreasing $K_{\mathrm{H,L}}$ below 0.2 mA does not allow for a self-sufficient network for the given topology.



(a) Optimal system timings (b) Optimal battery size

Fig. 3. Optimal system configuration parameters obtained for a specific topology ($n_1 = 5, H = 5, \xi = 1.5, \alpha = 1.5$) and varying $K_{H,L}$, the average energy income.

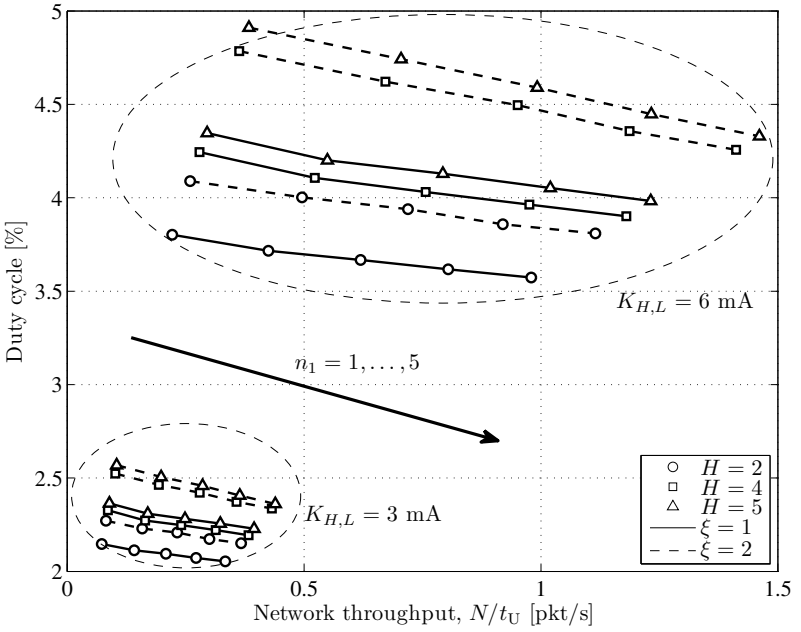


Fig. 4. Optimal system performance obtained for two energy income conditions (summer time, $K_{H,L} = 6$ mA, and winter time $K_{H,L} = 3$ mA) varying the network topology ($n_1 \in [1, \ldots, 5], H \in [2, 4, 5], \xi \in [1, 2], \alpha = 1.5$).

Fig. 3(b) shows the needed battery size vs I_{H}. Note that, due to (5), the optimal battery size is capped by the maximum energy that the bottleneck node spends when using the channel at its full capacity. Moreover, the different cap levels are due to the different optimal t_{U} and t_{off} when varying $K_{\mathrm{H,L}}$.

Finally, Fig. 4 shows results obtained for S2: here the aggregate network throughput, N/t_{U}, is plotted against the duty cycle, $t_{\mathrm{on}}/(t_{\mathrm{on}} + t_{\mathrm{off}})$, varying the network topology. In particular, curves are grouped in two sets: the first, in the top right part of the picture is related to $K_{\mathrm{H,L}} = 6$ mA, the second, in the bottom left part is obtained for $K_{\mathrm{H,L}} = 3$ mA. Solid lines are obtained for $\xi = 1$, while dashed lines represents the case $\xi = 2$. Round, square and triangular markers show results for network depths of 2, 4, and 5 hops, respectively. Each curve is plotted varying n_1 from 1 (leftmost point) to 5 (rightmost point).

Note that the network throughput increases with n_1, H and ξ because the impact of an increasing N dominates over that of the corresponding decreasing t_{U}. Instead, the duty cycle increases with H and ξ, but slightly decreases with n_1. Also, Fig. 4 shows that with a very small average energy income $K_{\mathrm{H,L}} = 6$ mA (e.g., by small photo-voltaic panels), it is possible to obtain a self sufficient network of 155 nodes, each of them sending packets every 2 minutes for an aggregate throughput of about 1.5 packet per second using a duty cycle of about 4.5 % (topology with $n_1 = 5$, $H = 5$, $\xi = 2$).

4 Discussion and Future Work

In this paper we have derived an energy consumption model for networks of embedded devices based on the network topology and the main communication procedures at the bottleneck nodes. Thus, we have introduced a second energy production model to represent renewable energy sources, such as photo-voltaic panels. Combining the two models we have obtained a constrained non-linear optimization problem which is solvable through interior point algorithms.

The solutions of this optimization problem provides the optimal configuration parameters for the perpetual operation of a network of embedded devices powered by energy harvesters. Even though our results have been obtained for a specific communication solution, we note that the analysis is promptly adaptable to different protocols and hardware technologies, making the proposed framework a versatile method to obtain optimal configuration parameters and performance bounds.

Based on these results, network protocols can be adapted on the fly by selecting the optimal information update time, t_{U}, and the optimal duty cycle, $t_{\mathrm{on}}/(t_{\mathrm{on}} + t_{\mathrm{off}})$, based on the current energy income rate. This makes networked embedded devices not only self-sufficient, but also flexible to fluctuations in the amount of energy harvested.

Possible avenues for future research include obtaining close form expressions for the optimal parameters, and a more accurate model for the multiple access interference, that shall account for retransmission dynamics and packet collisions. In particular, the latter extension would allow for a more precise estimation of the energy expenditure for a given network configuration in the high data traffic regime. Finally, the characterization of the energy buffer dynamics would allow the design of adaptation rules which certain guarantees in terms of buffer depletion probability, which may occur due to sudden variations in the amount of energy harvested.

References

1. Bonetto, R., Bui, N., Rossi, M., Zorzi, M.: McMAC: a power efficient, short preamble multi-channel medium access control protocol for wireless sensor networks. In: ICST/IEEE SIMUTools, Desenzano, Italy (March 2012)
2. Boyd, S., Kim, S.J., Vandenberghe, L., Hassibi, A.: A tutorial on geometric programming. Springer Optimization and Engineering 8(1), 67–127 (2007)
3. Buettner, M., Yee, G.V., Anderson, E., Han, R.: X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In: ACM SenSys. Boulder, Colorado, USA (October 2006)
4. Bui, N., Castellani, A.P., Casari, P., Rossi, M., Vangelista, L., Zorzi, M.: Book Chapter: Implementation and performance evaluation of wireless sensor networks for smart grids. Cambridge University Press (2012)
5. Dulman, S., Rossi, M., Havinga, P., Zorzi, M.: On the hop count statistics for randomly deployed wireless sensor networks. International Journal of Sensor Networks 1(1/2), 89–102 (2006)
6. Ko, J., Gnawali, O., Culler, D., Terzis, A.: Evaluating the Performance of RPL and 6LoWPAN in TinyOS. In: Workshop on Extending the Internet to Low Power and Lossy Networks (IP+SN), Chicago, Illinois, USA (April 2011)
7. Leong, J., Culler, D.: A Practical Theory of Micro-Solar Power Sensor Networks. ACM Transactions on Sensor Networks 9(1) (April 2012)
8. Liu, R.S., Fan, K.W., Zheng, Z., Sinha, P.: Perpetual and Fair Data Collection for Environmental Energy Harvesting Sensor Networks. IEEE/ACM Transactions on Networking 19(4), 947–960 (2011)
9. Michelusi, N., Stamatiou, K., Zorzi, M.: On optimal transmission policies for energy harvesting devices. In: Information Theory and Applications Workshop (ITA), San Diego, California, USA (February 2012)
10. Vullers, R., van Schaijk, R., Doms, I., Van Hoof, C., Mertens, R.: Micropower energy harvesting. Elsevier Solid-State Electronics 53(7), 684–693 (2009)

Experiment Design for Parameter Estimation in Sensing Models

Vladimir V. Shakhov

Institute of Computational Mathematics and Mathematical Geophysics
Siberian Branch of Russian Academy of Science
Novosibirsk 630090, Russia
shakhov@rav.sscc.ru

Abstract. In this study, the problem of quality improvement for sensing models in wireless sensor networks is considered. A choice of sensing model strongly influences design of wireless sensor networks and performance of protocols, such as traffic aggregation, broadcasting, energy efficient routing, target or barrier coverage etc. For these reasons, it is useful to investigate the nature of sensing ability, and the manner in which it depends on the characteristics of wireless sensor networks. The author investigates the impact of experiment design on the evaluation of a sensing quality function. The goal of this work is to provide an approach for the statistically efficient estimation of sensing model parameters.

Keywords: Wireless Sensor Networks, Sensing Model, Experiment Design.

1 Introduction

One of the most important performance measures of wireless sensor networks is the quality of tools required to detect a considered event. The sensing ability of wireless sensors is described by a sensing model. A choice of sensing model strongly influences design of wireless sensor networks and performance of protocols, such as traffic aggregation, broadcasting, energy efficient routing, target or barrier coverage etc. For these reasons, it is useful to investigate the nature of sensing ability, and the manner in which it depends on the characteristics of wireless sensor networks. The sensing model assumptions influence directly the whole wireless sensor network design, and it would be very difficult (sometimes, impossible) to develop efficient wireless sensor networks without adequate sensing models.

Let us provide clarification for the previous proposition. The probabilistic detection approach is the prevailing methodological framework for modeling sensing range. Its use often requires some simplifying assumptions, however these models provide a basis for adequate sensing range approximations, as well as valuable results and rewarding insights. It is a well known fact that a cost of sensor components is a critical consideration in the design of practical sensor networks. A cost of sensor network increases with sensor battery power. In conventional wireless sensor networks it is often economically advantageous to discard

G. Bianchi, A. Lyakhov, and E. Khorov (Eds.): WiFlex 2013, LNCS 8072, pp. 151–158, 2013.

a sensor rather than sensor recharging. By this reason a battery power is usually a scare component in wireless sensors. On the other hand, sensing range depends on battery productivity. Therefore, the factor of loss in detection process cannot be ignored. It means that inadequate sensing models lead to the overcharge or undercharge of sensor cost (or the number of sensors in the network), inefficient management of sensors duty cycles, inefficient sensor allocation, etc.

As it was mentioned above, the sensor detection ability is generally evaluated probabilistically. The probability of event detection degrades with distance between the sensor and the target. Generally, a function of detection probability (a sensing model) is specified. However, it contains an unknown parameter, which is has to be experimentally evaluated. The parameter depends on concrete situations, signal propagation conditions etc. The proper choice of the sensing model parameters defines the sensing model adequacy. In this study, the author investigates the impact of experiment design on the estimation of the mentioned parameter.

The rest of the paper is organized as follows. In Section 2 we discuss existing sensing models and demonstrate the impact of model parameter accuracy on wireless sensor network performance. Some details of experiment design technique are provided as well. In Section 3 we consider the problem statement for an abstract sensing model. In Section 4 we consider a wide used probabilistic sensing model and provide the optimal experiment plan. Finally, we conclude the paper in Section 5.

2 Preliminaries

2.1 Sensing Models

In the literature, a few types of sensing quality models are described. First one is the binary sensing model [1]. Accordingly the binary sensing model the detection ability of sensor is defined as follows

$$P = \begin{cases} 1, & \text{if } x < R \\ 0, & \text{otherwise.} \end{cases}$$

Here and below x is the Euclidean distance between the sensor and the event. R is the sensing range of this sensor, it is generally given. The model can be used in some applications. However, the binary model is not adequate in most cases. The most popular model is the probabilistic sensing model [2–6].

$$P = \begin{cases} 1, & \text{if } x \le R \\ e^{-a(x-R)}, & \text{otherwise.} \end{cases}$$

Now the sensing range R expresses the detection area inside which a sensor is possible to detect an event without loss. The positive parameter α is used to evaluated the quality of the sensor detection outside the sensing range R. Typically

the value of α is dependent on environmental and sensor characteristics. The parameter has to be obtained though experiments [6] and the optimal experimental design technique needs to be used. Remark that the area covered by a wireless sensor network can be heterogeneous and the parameter can get different values within the same network. In this case a series of distributed observations is required, the experiment cost grows drastically, and optimal experiment design becomes a very important and urgent problem.

Let us demonstrate the impact of model parameter estimation accuracy on wireless sensor network performance and provide the corresponding numerical examples. Consider, without loss of generality, the probabilistic sensing model in which $R = 0$. Let a group of sensors have to be used to detect some event. It is reasonable to reduce the system cost, i.e. the number n of sensors, providing that the detection probability is not less than a pre-defined threshold q. Assume, the distance from each sensor to the monitored target is the same and equals x. The probability of event detection is calculated as follows

$$P(n) = 1 - (1 - e^{-\alpha x})^n.$$

To provide probability $P(n)$ greater than or equal to q the required number of sensors is

$$n \geq \frac{\ln(1-q)}{\ln(1-e^{-\alpha x})}.$$

Let $q = 0.99, x = 2, \alpha = 1$. Therefore, the optimal number of sensors equals 32. If the value α is underestimated then the sensing quality requirement can be violated. If the estimated $\alpha = 1.1$, then the calculated n is equal 40. Therefore, if the relative error in the parameter estimation is 10 percent then 25 percent of excess sensors are applied. Thus, the cost of network deployment is essentially overstated. And not only that, the communications cost is groundlessly increased as well.

Let us provide one more example and consider wireless sensors with duty cycle. Assume, it is enough to use one sensor for target coverage if the detection probability is greater than or equal to q. In this way redundant sensors pass to sleeping mode and wait for their duty cycle slots. Let the sensors distribution density in the monitored area be constant. The number of sensors distributed in some region is directly proportional to the square of this region. A sensor is used for the target coverage if the distance between the sensor and target is as follows

$$x \leq -\frac{\ln q}{\alpha}.$$

If $q = 0.9$, the real $\alpha = 1$ and the estimated value of α is equal to 1.1 then the square of field containing the applicable sensors is unreasonably reduced. In this example, the estimated number of sensor for target coverage is decreased by 17 percent, comparing with the real one. Hence, the duty cycle is reduced by 17 percent and the network lifetime degrades significantly.

In the literature some alternative sensing quality models had been offered (see, for example [7]). It is generally accepted that the sensing quality function decreases nonlinearly with increase of the sensor-event distance. This property will be used below. The aim of this work is to investigate experiment conditions for the statistically efficient estimation of parameters of a sensing quality function.

2.2 Experiment Design Technique

Assume that the sensing quality is described by the following regression model

$$\eta(x, \vartheta) + e,$$

where ϑ is a set of unknown parameters, and e is a random error term. The regression function is known up to parameters ϑ. Using N observations it needs to estimate the parameters.

Let us define the experimental design $\varepsilon(N)$. It is the following set

$$\varepsilon(N) = \begin{pmatrix} x_1, \ldots, x_n \\ r_1, \ldots, r_n \end{pmatrix},$$

where r_i represents the number of experiment observations executed at the design point x_i, n is a total number of points,

$$\sum_{i=1}^{n} r_i = N.$$

The Fisher information matrix M is obtained as follows

$$M(\varepsilon(N)) = \sum_{i=1}^{n} \omega(x_i) r_i (\nabla\eta(x, \vartheta))^T \nabla\eta(x, \vartheta),$$

where $\omega(x)$ is the inverse response variance function.

If the regression coefficients are determined by the least squares method then the matrix M^{-1} equals the covariance matrix of ϑ estimations. Therefore, a choice of experiment design defines the quality of model parameters. In the literature, a different criteria of optimal design have been formulated. The most commonly used approach is the maximization of the determinant of M. The corresponding design is named as the D-optimal design. The design minimizes the generalized variance of the estimated parameters. Corresponding theoretical justifications can be found in [8].

If $\eta(x, \vartheta)$ is not linear in ϑ, then the matrix M depends on ϑ. A priori information of ϑ can be used for design improvement. However in this case the optimal design procedure cannot be provided. Thus, it is useful to get the optimal design in explicit form, even in particular cases. The corresponding results for sensing models is presented in the next section.

3 D-Optimal Design for Sensing Models

Let us consider one-parameter regression sensing models. In this case the determinant of Fisher information matrix takes the form

$$\det M(\varepsilon(N)) = \sum_{i=1}^{n} \omega(x_i) r_i \left(\frac{\partial \eta(x, \vartheta)}{\partial \vartheta} \right)^2.$$

The D-optimal design ε^* is obtained as the solution of the following optimization problem

$$\max \det M(\varepsilon(N)),$$

$$\sum_{i=1}^{n} r_i \leq N,$$

$$x \in \Omega_x,$$

Here Ω_x is the design space (the acceptable region for x). Let us receive the D-optimal design. Remark that

$$\det M(\varepsilon(N)) \leq \max_{x \in \Omega_x} \omega(x) \left(\frac{\partial \eta(x, \vartheta)}{\partial \vartheta} \right)^2 \sum_{i=1}^{n} r_i.$$

Therefore,

$$\det M(\varepsilon(N)) \leq N \max_{x \in \Omega_x} \omega(x) \left(\frac{\partial \eta(x, \vartheta)}{\partial \vartheta} \right)^2,$$

which implies

$$\varepsilon^* = \begin{pmatrix} x_1 \\ N \end{pmatrix}, x_1 = \arg \max_{x \in \Omega_x} \left(\omega(x) \left(\frac{\partial \eta(x, \vartheta)}{\partial \vartheta} \right)^2 \right).$$

The optimal design contains one point. The following equation is used to obtain the point

$$\frac{\omega'_x(x)}{\omega(x)} = -\frac{\eta''_{\vartheta x}}{\eta'_\vartheta}.$$

Taking into account the specificity of wide used sensing models, it can be assumed that $(\eta'_\vartheta(x, \vartheta))^2$ is a monotonic decreasing function of x. Assume, $\omega(x)$ is a positive constant.

From here

$$x_1 = \arg \max_{x \in \Omega_x} \left| \frac{\partial \eta(x, \vartheta)}{\partial \vartheta} \right| = \min\{x : x \in \Omega_x\}.$$

Let us remark that the set Ω_x can be discrete. For example, it takes place if observations executers and targets are proper sensors of network and the design

space is limited by network sensors. Therefore, experiment runs for sensing quality estimation of selected sensor has to be executed by the nearest sensor. Under the probabilistic sensing model above,

$$x_1 = \min\{x : x \in \Omega_x, x > R\}.$$

If an area covered by the considered wireless sensor network is assumed to be homogeneous (i.e. the model parameter is the same for all sensors) then the estimation of probabilistic sensing model parameter for all sensors has to be done with using of the couple of closest sensors.

4 Performance Evaluation

Let us consider the probabilistic sensing model

$$\eta(x, \vartheta) = e^{-\alpha x}, x \in (0, \infty).$$

Assume, $\alpha = 1, \omega(x) = \text{const}, x_1 = 1$, the number of experiment runs equals 100. Let us provide the performance comparison between the optimal design

$$\varepsilon^* = \begin{pmatrix} x_1 = 1 \\ N = 100 \end{pmatrix},$$

and the following even design

$$\varepsilon(N) = \begin{pmatrix} x_1, \ldots, x_n \\ r_1, \ldots, r_n \end{pmatrix}, x_i = 1 + (i-1)s, i = 1 \ldots N,$$

here s is a variable parameter. Fig.1 provides the ratio of estimation variances, i.e. $D(\varepsilon^*)/D(\varepsilon(N))$, as a function of s. Here we take into account the following fact

$$\frac{D(\varepsilon^*)}{D(\varepsilon(N))} = \frac{\det M(\varepsilon(N))}{\det M(\varepsilon^*)}$$

It is shown that the estimation quality rapidly decreases as the distance between experiment points of $\varepsilon(N)$ increases. The optimal plan produces estimates, which are ten times better than estimates based on $\varepsilon(N)$.

Assume, $\omega(x) = x$. Therefore,

$$\det M(\varepsilon^*) = Nxe^{-2\alpha x}.$$

The optimal point is defined as

$$x_1 = \frac{1}{2\alpha}.$$

The optimal design depends on unknown parameter. However, if we substitute the expression for x_1 in the regression equation then we get

$$\eta(x, \alpha) = e^{-0.5} \approx 0.61,$$

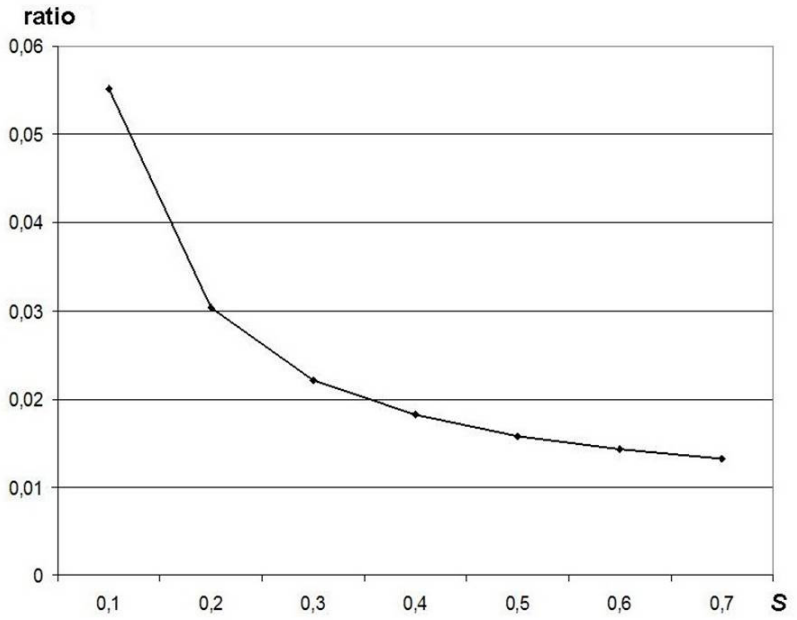


Fig. 1. The ratio of estimation variances

it does not depend on the parameter. Preliminary experiments can be used to evaluate the distance, where the loss rate gets 39 %. And the optimal design ε^* has to be concentrated around this level. Assume, $\alpha = 1$. Let us consider the alternative experiment design ε_2 focused at the point $x_2 = 0.105$. The corresponding detection rate is about 90 %. We obtain

$$D(\varepsilon_2)/D(\varepsilon^*) = \frac{x_1}{x_2} \exp(2\alpha(x_2 - x_1)) \approx 2.2.$$

Thus, in this case the proposed approach allows to improve the estimation quality as well.

5 Conclusion

In this paper, we investigate experiment conditions for sensing model parameters estimating and offer an approach for optimal experiment design computing. The D-optimal design criterion is considered. For the considered cases it coincides with the A-optimal design criterion (maximization of the trace of the Fisher information matrix [8]). For the wide used sensing model it is shown that the optimal design has to be concentrated in one point of the design space. The method of the point calculation is offered. The proposed approach being implemented allows to minimize experiments costs for sensing quality estimation, reduce the network traffic related to the considered experiments, improve energy consumption for wireless sensor networks.

Acknowledgments. The research was supported in part by the Russian Foundation for Basic Research (grant 11-07-00183).

References

1. Chakrabarty, K., Iyengar, S., Qi, H., Cho, E.: Grid coverage for surveillance and target location in distributed sensor networks. IEEE Transactions on Computers 51(12), 1448–1453 (2002)
2. Dhillon, S., Chakrabarty, K.: Sensor placement for effective coverage and surveillance in distributed sensor networks. In: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2003), New Orleans, pp. 1609–1614 (2003)
3. Zou, Y., Chakrabarty, K.: Sensor deployment and target localization based on virtual forces. In: Proceedings of the Annual Joint Conference of the Computer and Communications Societies, pp. 1293–1303. IEEE Press (2003)
4. Lin, Z., Zhang, S., Yan, G.: An incremental deployment algorithm for wireless sensor networks using one or multiple autonomous agents. Ad Hoc Networks 11(1), 355–367 (2013)
5. Altinel, I., Aras, N., Gney, E., Ersoy, C.: Binary integer programming formulation and heuristics for differentiated coverage in heterogeneous sensor networks. Computer Networks, 2419–2431 (2008)
6. Yang, Q., He, S., Li, J., Chen, J., Sun, Y.: Energy-Efficient Probabilistic Full Coverage in Wireless Sensor Networks. In: Proceedings of IEEE Globecom, USA, pp. 609–614 (2012)
7. Megerian, S., Koushanfar, F., Qu, G., Veltri, G., Potkonjak, M.: Exposure in wireless sensor networks: Theory and practical solutions. Wireless Networks 8(5), 443–454 (2002)
8. Pukelsheim, F.: Optimal Design of Experiments. John Wiley & Sons, Inc., New York (1993)

Author Index

Abdul Latiff, N.M. 125
Asadpour, V. 125
Asheralieva, Alia 1

Bikov, Evgeni 13
Botvich, Dmitri 37
Bui, Nicola 138

Doronin, Dmitry 112

Elizarov, Stanislav 13

Fakhriev, Denis 112
Fisal, N. 125
Frolov, Alexey 49

Gallo, Pierluigi 80
Guschin, Andrey 69

Khan, Jamil Y. 1
Khorov, Evgeny 69, 93
Kiryanov, Anton 69
Krasilov, Artem 93

Loziak, Krzysztof 58
Lyakhov, Andrey 69, 93

Mahata, Kaushik 1
Malone, David 37

Namiot, Dmitry 25
Natkaniec, Marek 58
Nejatian, S. 125

Osipov, Dmitry 49
Ostrovsky, Dmitry 93

Patras, Paul 37

Qi, Hanghang 37

Rossi, Michele 138

Safonov, Alexander 69
Shakhov, Vladimir V. 151
Sikora, Marek 58
Sneps-Sneppe, Manfred 25
Syed-Yusof, S.K. 125
Szott, Szymon 58

Tinnirello, Ilenia 80

Zyablov, Victor 49