JYVÄSKYLÄ STUDIES IN COMPUTING
143

Vitaliy Tykhomyrov

Mitigating the Amount of Overhead Arising from the Control Signaling of the IEEE 802.16 OFDMa System

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 143

Vitaliy Tykhomyrov

Mitigating the Amount of Overhead Arising from the Control Signaling of the IEEE 802.16 OFDMa System

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella julkisesti tarkastettavaksi yliopiston Agora-rakennuksen auditoriossa 1 joulukuun 15. päivänä 2011 kello 12.

Academic dissertation to be publicly discussed, by permission of the Faculty of Information Technology of the University of Jyväskylä, in the building Agora, auditorium 1, on December 15, 2011 at 12 o'clock noon.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2011

Mitigating the Amount of Overhead Arising from the Control Signaling of the IEEE 802.16 OFDMa System

JYVÄSKYLÄ STUDIES IN COMPUTING 143

Vitaliy Tykhomyrov

Mitigating the Amount of Overhead Arising from the Control Signaling of the IEEE 802.16 OFDMa System

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2011

Editors
Timo Männikkö
Department of Mathematical Information Technology, University of Jyväskylä
Pekka Olsbo, Ville Korkiakangas
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-4552-7
ISBN 978-951-39-4552-7 (PDF)
ISSN 1456-5390

Jyväskylän yliopisto, 2023

ISBN 978-951-39-4551-0 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2011

ABSTRACT

Tykhomyrov, Vitaliy
Mitigating the Amount of Overhead Arising from the Control Signaling of the IEEE 802.16 OFDMa System
Jyväskylä: University of Jyväskylä, 2011, 52 p.(+included articles)
(Jyväskylä Studies in Computing
ISSN 1456-5390; 143)
ISBN 978-951-39-4551-0 (nid.)
ISBN 978-951-39-4552-7 (PDF)
Finnish summary
Diss.

This work focuses on mitigating the amount of overhead arising from the control signaling of the IEEE 802.16 system. It analyses existent solutions and proposes efficient methods and algorithms. Although existent IEEE 802.16 standard proposes some techniques, it is worthwhile to point out that exact implementation is not specified and left open for alternative implementations. The first part of this work addresses the main aspects of MAC layer overhead. The second part of this work introduces efficient techniques to improve the performance of ARQ mechanism. It is shown that the ARQ mechanism can improve significantly the performance of TCP based applications and confirms that a correct configuration plays an important role in transmitting data over wireless channels in the IEEE 802.16 OFDMa system. It is also of the view that the proposed solutions in this work must be verified by means of extensive simulations, and therefore each proposed method and algorithm is implemented and tested within the WINSE simulator in the NS-2 framework.

Keywords: ARQ, NS-2, overhead, VoIP, WiMAX, WINSE

Author Vitaliy Tykhomyrov
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Supervisors Prof. Timo Hämäläinen
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Dr. Alexander Sayenko
Research, Technology & Platforms
Nokia Siemens Networks
Finland

Reviewers Assoc. Prof., Dr. Tien Van Do
Department Of Telecommunications
Budapest University of Technology and Economics
Hungary

Prof. János Sztrik
Department of Informatics Systems and Networks
Faculty of Informatics
University of Debrecen
Hungary

Opponent Prof. Victor Netes
Moscow Technical University of Communications and Informatics
Russia

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisors Timo Hämäläinen and Alexander Sayenko for their precious suggestions and guidance during the writing of this dissertation. Their guidance and inspiration have provided an invaluable experience that will help me in my career. I would never have completed this work without their continuous encouragement and could not have wished for better supervisors. My sincere appreciation goes to the Department of Mathematical Information Technology and Jyväskylä Graduate School in Computing and Mathematical Sciences for the technical and financial sponsorship.

I want to express my gratitude to Assoc. Prof. Tien Van Do and Prof. János Sztrik for the review of this dissertation and Prof. Victor Netes for being my opponent.

I would like to thank my numerous colleagues and friends who helped me throughout this academic exploration. Several people have played a decisive role in saving me from leaving research in telecommunications. These persons are professor Vladimir Sayenko (Kharkiv National University of Radio and Electronics) and Oleksiy Grytsenko. Without your willingness and suggestions, this academic exploration would not have been started. Special acknowledgements must be given to the reviewers of various conferences and journals for the thorough review and constructive remarks for ensuring a high technical quality level of my articles.

On a personal note, I wish to thank my dearest and beloved wife Anytka for her love, support, patience and understanding during these years. Last but not least, my love goes to my parents and all my relatives for their love and support.

Jyväskylä, October 26, 2011,
Vitaliy Tykhomyrov

GLOSSARY

3G	3rd Generation
3GPP	3rd Generation Partnership Project
ACK	Acknowledge
ARQ	Automatic Repeat Request
BE	Best Effort
BS	Base Station
CBR	Constant Bit-Rate
CDF	Cumulative Distribution Function
DL	Downlink
DL-MAP	Downlink Map
DSA	Dynamic Service Addition
DSC	Dynamic Service Change
DSL	Digital Subscriber Line
ertPS	Extended Real-time Polling Service
E2E	End-to-End
FCH	Frame Control Header
FPS	Frames Per Second
FTP	File Transfer Protocol
FUSC	Fully Used Sub-Channel
HARQ	Hybrid Automatic Repeat Request
IE	Information Element
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
LTE	Long Term Evolution
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
NACK	Not Acknowledge
NGMN	Next Generation Mobile Networks
nrtPS	Non-Real-Time Polling Service
NS-2	Network Simulator v2
NS-3	Network Simulator v3
OFDM	Orthogonal Frequency Division Multiplex
OFDMa	Orthogonal Frequency Division Multiple Access
PDU	Protocol Data Unit
PHY	Physical Layer
PSH	Packing Sub-header
PUSC	Partially Used Sub-Channel
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RTG	Receive/transmit Transition Gap

rtPS	Real-Time Polling Service
SDU	Service Data Unit
SS	Subscriber Station
TCP	Transmission Control Protocol
TDD	Time Division Duplex
TTG	Transmit/receive Transition Gap
UDP	User Datagram Protocol
UGS	Unsolicited Grant Service
UL	Uplink
UL-MAP	Uplink Map
VBR	Variable Bit-Rate
VoIP	Voice over IP
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network

LIST OF FIGURES

LIST OF TABLES

CONTENTS

LIST OF INCLUDED ARTICLES

PI A. Sayenko, V. Tykhomyrov, H. Martikainen and O. Alanen. Performance Analysis of the IEEE 802.16 ARQ Mechanism. *The 10-th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2007.

PII V. Tykhomyrov, A. Sayenko, H. Martikainen, O. Alanen and T. Hämäläinen. Performance evaluation of the IEEE 802.16 ARQ mechanism. *7th International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking*, 2007.

PIII Vitaliy Tykhomyrov, Alexander Sayenko, Henrik Martikainen, Olli Alanen and Timo Hämäläinen. On ARQ feedback intensity of the IEEE 802.16 ARQ mechanism. *International Conference on Telecommunications*, 2008.

PIV Vitaliy Tykhomyrov, Alexander Sayenko, Henrik Martikainen and Olli Alanen. Analysis and Performance Evaluation of the IEEE 802.16 ARQ Mechanism. *Journal of Communications Software and Systems, vol 4, issue 1*, 2008.

PV Oleksandr Puchko, Vitaliy Tykhomyrov and Henrik Martikainen. Link Adaptation Thresholds for the IEEE 802.16 Base Station. *The Second Workshop on NS-2*, 2008.

PVI Alexander Sayenko, Olli Alanen, Henrik Martikainen, Vitaliy Tykhomyrov, Oleksandr Puchko, Vesa Hytönen, Timo Hämäläinen. WINSE: WiMAX NS-2 Extension. *Special Issue of Simulation: Software Tools, Techniques and Architectures for Computer Simulation*, 2010.

PVII V. Tykhomyrov and A. Sayenko and O. Puchko and T. Hämäläinen. Decreasing the MAP overhead in the IEEE 802.16 OFDMa system. *European Wireless Conference*, 2010.

PVIII Vitaliy Tykhomyrov. Increasing the VoIP capacity through MAP overhead reduction in the IEEE 802.16 OFDMa Systems. *Mobile Lightweight Wireless Systems: Second International ICST Conference*, 2010.

1 INTRODUCTION

In the last few years the demand for broadband connections has grown remarkably. Unfortunately, in some areas that are difficult to reach cable solutions are too expensive. Obtaining high data rates, with certain QoS requirements, is quite important. Here the IEEE 802.16 standard takes a relevant role in implementing a full-service network. The standard provides high data rates in a big covering range with low implementation costs and with multi-traffic communications. The IEEE 802.16 specifications were designed with the focus on flexibility, thus leaving several parts of it as optional and allowing various Base Station (BS) and Mobile Station (MS) implementations. Because the IEEE only sets specifications but does not test the equipment for compliance, the WiMAX Forum runs a certification program, which comprises more than 300 companies. WiMAX certification is intended to guarantee compliance with the standard and interoperability with equipment from other manufacturers. The mission of the forum is to promote and certify compatibility and interoperability of broadband wireless products. According to the latest WiMAX Forum statistics [1], the IEEE 802.16-based networks are deployed in 150 countries, totaling 583 separate deployments. In February, 2011, the WiMAX Forum announced that, as of the end of 2010, WiMAX service providers covered more than 823 Million people. This estimate surpasses last year's forecast of 800M people, and signifies that WiMAX [1] is still on target to reach the forecast that by 2011 there will be over 1 Billion people across the world within WiMAX coverage.

[1] WiMAX is not a technology, but rather a certification mark, given to equipment that meets certain tests for the IEEE 802.16 family of standards. This term has been adopted in popular usage to denote the technologies behind it. This is likely due to the difficulty of using terms like *IEEE 802.16* in common speech and writing. Using the terms *WiMAX* and *IEEE 802.16* interchangeably poses problems since it is feasible that equipment manufacturers could build products based on IEEE 802.16 standards that would not be labeled as WiMAX certified.

1.1 Motivation and objectives

The 802.16 standard includes only specifications for the Physical (PHY) and Medium Access Control (MAC) layers. This approach gives additional flexibility in the network management. It also means that different products that comply with the standard may have substantial differences in performance and in the implementation of the layers beyond the 802.16 MAC and PHY.

The IEEE 802.16e based systems definitely have a higher system capacity and a more sophisticated mechanism to provide a better quality of service(QoS) than previous wireless systems, such as code-division multiple-access(CDMA) [2],[3] or the universal mobile telecommunications system (UMTS) [4],[5]. A major concern with the 802.16 systems is the control of signaling overhead. When compared to 3G systems, the 802.16 has larger signaling structures due to more options, a more complicated frame structure, and a possibility to have a number of zones within a single frame with different permutation types. Another reason for the large signaling overhead is the scheduling and a need to ensure QoS requirements since the subscriber station (SS) cannot transmit data until it has been allocated a channel by the Base Station. The 802.16 standard was designed with a variety of traffic types in mind. It has to handle the requirements of very high data rate applications, such as voice over IP (VoIP) and video or audio streaming, as well as low data rate applications, such as web surfing, and handle extremely bursty traffic over the Internet. And it may need to handle all of these at the same time.

With non-real-time traffic, the BS scheduler is flexible in deciding when resources should be allocated and how much of them should be granted. However, the BS scheduler cannot delay allocating resources without ensuring the QoS requirements. On the other hand, there are a few key areas where the performance of 802.16 systems can be improved. Voice over IP (VoIP) capacity is one of them. The VoIP capacity of any 802.16 OFDMa system is related to the associated signaling overhead. Overhead is important for such applications due to the frequent transmission and small packet size. In the 802.16 system, much of the overhead associated with VoIP traffic occurs in MAP messages, since dynamic scheduling is used to support VoIP. It is worth noting that informing subscriber stations about new downlink or uplink resource assignments creates a signaling overhead that influences the system throughput. However, previous research has not taken the 802.16 signaling overhead adequately into consideration.

The Mobile WiMAX [6] and 3GPP-LTE [7] standards are technically similar [8],[9],[10]. However, from the perspective of the market these two standards differ in terms of legacy. Following this observation, it can be concluded that, due to timeline benefits, new service providers as well as existing cable and DSL providers wishing to offer mobile services are likely to select Mobile WiMAX as their technology for mobile broadband access. Major UMTS/HSPA service providers will evolve to 3GPP-LTE, and most CDMA2000 providers, as well as GSM/EDGE providers, will select Mobile WiMAX for mobile broadband wireless access while providing service continuity over their legacy networks.

1.2 Problem statement

The amount of overhead arising from control signaling is caused mostly by the MAP messages that carry information on data burst allocations. One of the major problems is that the 802.16 standard defines some mechanisms to decrease the MAP overhead, but does not provide concrete values and solutions. This means that it is left open for alternative implementations. Failing to set up the correct network parameters may result in decreased system performance. These issues are studied in Chapters 2 and 3.

Another problem is that it is quite difficult to accept a contribution if it is not based on reliable simulation results performed by researchers and engineers at various architectural levels. Some 802.16 simulators have been implemented by academia and industry [11],[12],[13],[14], [15]. However, most of them do not contain all the main features of the IEEE 802.16 standard or concentrate on some particular problem. Therefore, it is difficult to cover all the main aspects of the 802.16 standard.

This work analyses existent solutions and proposes efficient methods and algorithms for mitigating the amount of overhead arising from the control signaling of the IEEE 802.16 OFDMa system. Each proposed method and algorithm is implemented and tested within the WINSE extension in the NS-2 simulator. Also, some practical issues related to initial network setup are discussed.

1.3 Related research works

Reduction of signaling overhead has not been studied thoroughly in terms of how it affects the system performance, especially in the IEEE 802.16 OFDMa networks.

In [16], the authors propose a new optimization model and evaluate the performance of a dynamic OFDMa system with inband signaling in terms of how the signaling overhead affects the system performance. However, they consider the signaling overhead only in the downlink direction, even though the signaling overhead of both the downlink and uplink affects the system throughput.

In [17], the authors present an analysis of a Downlink Data Mapping algorithm for IEEE 802.16e OFDMa systems called "Mapping with Appropriate Truncation and Sort". The algorithm maps data, which is allocated by the downlink scheduling algorithm, to the appropriate rectangular regions in the time and frequency domain. As a part of this algorithm, the authors try to reduce the MAP overhead. However, the simulation setup is not clear. Moreover, some key issues, such as error modeling, are not taken into account.

In [18], the authors demonstrate an advanced scheme for reducing the portion of the radio resource that is used for MAP message transmission. They propose to transmit information on the allocations in the next frame by using a conventional data burst scheduled for the current frame. However, the proposed scheme uses a feedback mechanism to assure the transmission reliability of piggybacked information and focuses mainly on the MAC overhead. Furthermore, the proposed scheme does not take into account error modeling and the simulation setup is not described.

In [19], the authors propose a further compressed MAP called *Tiny MAP*, where redundant elements are removed and other small modifications are introduced. Obviously, the results show that this solution can decrease the overhead and thereby increase the final system performance. However, this solution does not conform to the IEEE 802.16 standard since it requires some modifications which are not covered by the standard [20].

In [21], the authors address the problem of allocation representation of VoIP packets and propose an efficient uplink mapping scheme that decreases the size of the allocation map in the IEEE 802.16e OFDMa system. Moreover, they also present analytical and simulation models to evaluate the performance of the VoIP services. Their results show that the signaling overhead is greatly decreased when the proposed mapping scheme is used. However, the proposed mapping scheme does not conform to the IEEE 802.16 standard since it requires some modifications. Furthermore, their simulator implementation does not contain all the main features of the IEEE 802.16 standard, such as error modeling.

In [22], the authors consider an AMC scheme to reduce the MAP signaling overhead without explicit information on the channel condition. Indeed, the proposed scheme achieves the same coverage as the broadcast MAP while significantly enhancing the VoIP capacity. However, the authors do not describe their implemented simulation environment, and therefore there is no information on whether it does contain all the main features of the IEEE 802.16 standard. Moreover, the authors do not consider the IEEE 802.16m Evaluation Methodology Document [23]. In order to evaluate the MAP overhead, the authors use only 4 MCS levels, while the basic 802.16e system operates 11 different MCS levels [20].

In [24], the authors propose an almost overhead-free error control mechanism for the 802.16-based multi-hop relay networks. By means of simulations, the authors show that implemented mechanism can effectively reduce the redundant overhead of CRC-32 in WiMAX MAC PDU and still possesses the ability of early detection of and discarding the error and useless data during transmission. However, the proposed discrete error checking scheme assumes that the structure of MAC PDU must be modified.

In [25], the authors present three types of proposal on reduction of the overhead caused by the ARQ mechanism in WiMAX networks. The first proposed scheme transmits only negative acknowledgment (NACK) to inform the transmitter about the status of received blocks. The second proposed scheme is based on the first one, and in addition it utilizes a different type of block acknowledgment. Finally, the last proposal combines features of the conventional

ARQ (according to IEEE 802.16e) and of the first two proposed schemes. The authors show that the proposals reduce the overhead without negatively influencing packet delay. However, the authors use MATLAB to perform the link level calculations; the important MAC procedures are not considered.

While the above contributions are interesting and to some degree relevant in the context of this work, the architecture of the radio systems investigated by the authors considered above differs from that of the IEEE 802.16 standard. For this reason, the techniques that have been devised by them are not applicable here. However, the few authors who tried to test the proposed schemes generally obtained unclear results due to unoptimistic simulation environment.

1.4 Outline of the dissertation

The rest of this dissertation is arranged as follows:

Chapter 2 addresses the main aspects of signaling overhead of the IEEE 802.16 system. It presents the basics of the 802.16 MAP types as well as discusses issues related to MAC overhead. It also presents the proposed solutions. Each proposed solution is tested in WINSE [26] by means of extensive simulations. Chapter 3 gives an overview of ARQ mechanism in the IEEE 802.16 OFDMa system. In particular, it presents ARQ fragmentation/packing mechanisms, ARQ block usage, ARQ feedback types and ARQ timers. Chapter 4 provides the latest simulation results. Chapter 5 concludes the dissertation and suggests future research topics. Finally, all related publications are located at the end of this dissertation.

1.5 Main contribution

During the work on the subject of this dissertation, the author has produced and co-authored several publications.

Publication **PI** analyzes various ARQ parameters and studies their impact on the performance of the ARQ mechanism. The idea was to analyze the system performance with different parameters, such as ARQ feedback type, scheduling of ARQ feedbacks and retransmissions, ARQ transmission window size, ARQ block size, and ARQ block rearrangement. The work was co-authored, and the contribution of the author included the analysis of the results, implementation of the algorithm to build ARQ block sequences and consideration of a choice for the ARQ feedback type. The author also performed a number of simulation scenarios to study these parameters and how they impact the performance of application protocols. In Publication **PII**, a more extended research is done and new simulation results are presented. The contribution of the author is the same as Publication PI.

In Publication **PIII**, an analysis of the ARQ feedback intensity of the IEEE 802.16 ARQ mechanism is presented, the author is the main contributor for this publication, having implemented the basic framework on the WINSE NS-2 simulator extension. The author also investigated different approaches for low and high ARQ feedback intensity values and proposed an optimal set of values.

Publication **PIV** studies the key features and parameters of the 802.16 ARQ mechanism. The authors propose a solution on how to prioritize normal PDUs, ARQ feedbacks, and retransmissions. The author was in charge of the design and implementation of the algorithms to select the ARQ feedback type and to build block sequences for the cumulative+sequence feedback type. The author also performed extensive simulations and analyzed the obtained results.

Publication **PV** considers a link adaptation model and conducts a number of extensive simulation runs to find transition thresholds for the ARQ and HARQ retransmission mechanisms. The author was in charge of a separate overview of several cases with ARQ and HARQ retransmission mechanisms, of preparing a simulation framework and studying wireless MAC and PHY issues with NS-2.

Publication **PVI** presents an 802.16 extension for the NS-2 network simulator. The work considers upper PHY modeling, most of the features of the 802.16 MAC layer, as well as the QoS framework. The work is co-authored, and the contribution of the author includes the initial planning of the NS-2 extension, helping in the design of the MAC layer, the analysis of the results and background research for the other IEEE 802.16 extensions in the area. The author was also in charge of the creation of the theoretical part.

Publication **PVII** considers several solutions to minimize the signaling overhead in 802.16 systems. The author is the main contributor for this publication, having designed and implemented several algorithms to build sub-MAPs based on the number of bursts the BS scheduler allocates in a frame. The publication shows that the optimal algorithm is less computationally expensive when compared to the brute force approach. The author is also the main person responsible for analyzing the results.

Publication **PVIII** contributes to the evaluation of VoIP capacity with enabled sub-MAPs in the IEEE 802.16 OFDMa system. It tackles the signaling overhead as one of the main issues with a supporting VoIP service over 802.16 networks. The author, who is the main contributor for this publication, ran complex dynamic simulations with all the major radio resource management algorithms of the IEEE 802.16 system and the VoIP service and gives several recommendations on how to achieve a significant VoIP capacity improvement in the IEEE 802.16 OFDMa system.

2 SIGNALING OVERHEAD OF THE IEEE 802.16 SYSTEM

This chapter is organized as follows. Subchapter 2.1 addresses the main aspects of downlink MAC layer overhead, and subchapter 2.2 discusses issues related to uplink MAC overhead. The proposed solutions are presented in subchapter 2.3. Later, these solutions are verified by means of extensive simulations in Chapter 4.

2.1 Downlink Overhead

The 802.16 standard defines control overhead as a part of the frame structure [27] and supports several types of duplexing mode such as Frequency Division Duplex (FDD) and Time Division Duplex (TDD). FDD is inefficient in handling asymmetric data since traffic may only occupy a small portion of a channel bandwidth. On the other hand, TDD's spectral efficiency is higher than that of FDD and allows adjusting the DL/UL ratio dynamically. Also it handles both symmetric and asymmetric broadband traffic. Thus, TDD is selected as the main duplexing mode in this work. A more detailed analysis on duplexing schemes available in the 802.16 system was done by one of the coauthors in [28].

Generic IEEE 802.16 TDD frame structure is shown in Figure 1. Partitioned into downlink and uplink subframes, each frame begins with the transmission of the following mandatory control overhead:

- a preamble that is required and used for frame synchronization, channel state estimation, received signal strength, and signal-to-interference-plus-noise ratio (SINR) estimation. On the downlink (BS talking to a SS) synchronization is provided by a fixed preamble bit pattern transmitted at the beginning of each frame. Since the preamble transmits no actual data, its presence increases the size of the control overhead;
- frame control header (FCH), which contains the DL frame prefix and speci-

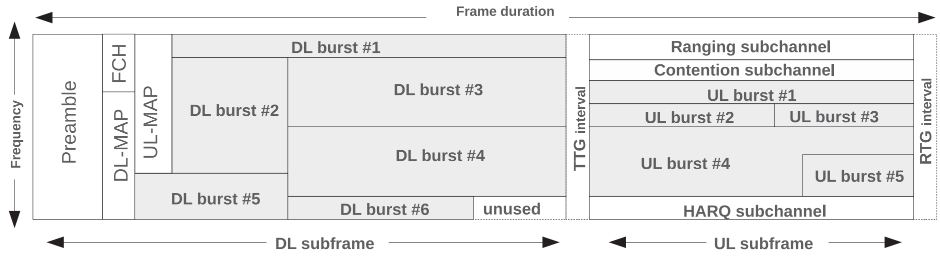


FIGURE 1 OFDMa TDD frame simplified structure.

fies the length and the repetition coding of the DL-MAP message.

- downlink map (DL-MAP) and uplink map (UL-MAP) messages that describe the structure and composition of the frame.
- Time gaps, i.e., transmit/receive transition gap (TTG) and receive/transmit transition gap (RTG), which are inserted between the downlink and uplink sub-frames, and at the end of each frame, respectively, in order to allow transitions between transmission and reception functions. This technique can increase the overhead slightly.

It is worth noting that FCH and DL-MAP are allocated within the DL sub-frame in column-wise order. The remaining part of the downlink sub-frame, as seen in Figure 1, is allocated as data regions. In OFDMA, the data region is a two-dimensional allocation of a group of contiguous subchannels, in a group of contiguous OFDMA symbols. All the allocations refer to logical subchannels. A two-dimensional allocation may be visualized as a rectangle of OFDMA slots. Each data region is specified through an Information Element (IE) into the DL-MAP. Therefore, the greater the number of data regions, the higher the total overhead due to the DL-MAP transmission.

Based on the packets waiting to be sent to various mobile stations, a BS determines the burst size in the downlink direction. Since the downlink is formed by one or more bursts of data, the SSs must have some way of keeping the timing of the bursts and knowing whether they need to listen to a given burst to receive their traffic. The information carries no user traffic and is considered as a part of the downlink channel overhead.

2.1.1 DL-MAP message

In the first broadcast burst following the FCH, the BS may insert a DL-MAP message to describe other bursts. DL-MAP is a MAC management message that contains information on allocation scheduled for downlink, the modulation and coding type, and the size and position of each allocation. There is one information element (IE) for each active connection. A connection is typically associated with one SS, but can be configured to be shared between multiple SSs.

Figure 2 shows the format of a DL-MAP message.

FIGURE 2 Format of DL-MAP message.

A broadcast message such as DL-MAP should be sent at the highest reliability needed to meet the cell edge coverage. Since DL-MAP is the most frequently delivered message, it should be included in every frame. If an erroneous MAP is received, then subscriber stations cannot find the location of bursts allocated to them. Thus, DL-MAP should be transmitted reliably using quite robust modulation and coding scheme (MCS) to ensure that all the stations receive them correctly. As a result of using robust MCSs, the control overhead increases significantly, which might become a performance bottleneck for certain applications, such as VoIP. In some systems, the MAP overhead might occupy up to half of the frame if there are short data bursts for many users to reside in a frame. How often DL-MAPs are sent depends on the channel configuration and how the BS schedules the traffic [29].

2.1.2 Compressed MAP message

The 802.16 MAP entries result in a significant amount of overhead since they are repeated four times. WiMAX Forum recommends using compressed MAP [30], which reduces the DL-MAP entry overhead to 11 bytes including 4 bytes for Cyclic Redundancy Check (CRC) [27].

Compressed MAP integrates DL-MAP and UL-MAP into a single message and saves space by removing several fields. Figure 3 shows the format of compressed MAP messages. As can be seen, there is a common CRC field, but no GMH and Type fields.

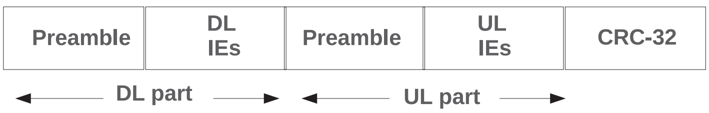


FIGURE 3 Format of compressed MAP message.

The only drawback of the compressed MAP message is that if it is dropped, then both the DL-MAP and UL-MAP entries are lost. However, similarly to the normal MAP message, the compressed MAP should be transmitted with quite a robust MCS to ensure that all the stations receive it correctly.

2.1.3 Sub-MAP message

In order to reduce the data bandwidth overhead in sending DL-MAP or UL-MAP, the allocation of data bursts can be achieved by utilizing different types of messages that can be transmitted using more efficient modulation coding schemes (MCSs). The 802.16 standard introduces sub-MAPs that allow splitting of a MAP message into a number of independent messages, each of which is encoded with a more efficient MCS. Sub-MAPs follow the compressed MAP as shown in Figure 4. The sub-MAP format is quite similar to the compressed MAP, with a few additional enhancements, such as the CRC-16 field instead of the CRC-32 field.

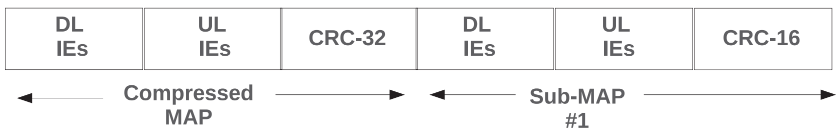


FIGURE 4 Compressed MAP with sub-MAP.

The only drawback of the sub-MAP message is that if it is dropped, then both the DL-MAP and UL-MAP entries are lost. However, similarly to the compressed MAP message, the sub-MAP should be transmitted with quite a robust MCS to ensure that all the stations receive it correctly. On the other hand, sub-MAPs with a more robust MCS will send less data when compared to other sub-MAPs with a more efficient MCS. Thus, choosing an optimal MCS is a tradeoff between robustness and efficiency. The current version of the 802.16 standard supports up to three sub-MAP messages.

2.2 Uplink Overhead

The uplink subframe also has fixed and variable parts. For example, ranging, contention, channel quality indication (CQI) and acknowledgements (ACK) reside in the fixed part. Their size is defined by the network operator. Hence, the bigger the number of slots allocated for the fixed part, the smaller the number of slots available for other data. Each uplink burst begins with a preamble whose length is defined by the BS through the UCD message. Typically, one OFDMA symbol is used for a short preamble and two for a long preamble. On the other hand, the uplink is shared between a collection of SSs, and the BS receiver similarly needs to synchronize with each SS transmitter when they begin to transmit a new burst of data consisting of one or more consecutive symbols in a frame. It tends to a situation where each SS prepends a preamble to the beginning of each burst. The preambles can occupy several symbols, as mentioned above, at the beginning of every burst. This overhead is variable depending on the burst size in terms of the number of symbols and depends on the type of user traffic that is being carried by the channel. In the worst case, SSs send bursts of very short data requiring only a single symbol per frame. As a solution, user traffic can be buffered to a group of data requests forming longer bursts.

Usually uplink data bursts are allocated as horizontal stripes. In this case the transmission starts at a particular slot and continues until the end of the uplink subframe and continues on the next subchannel. As a result, it minimizes the total number of subcarriers used by the SS and maximizes the power per subcarrier.

2.2.1 UL-MAP message

FIGURE 5 Format of UL-MAP message.

Figure 5 shows the format of UL-MAP message. Following the FCH and DL-MAP (when present), the UL-MAP message indicates how the upcoming uplink channel frame is allocated. An UL-MAP is necessary, since without it the SSs would have no way of managing their uplink access. The UL-MAP message (when present) is always transmitted in the first PDU on the burst described by the first DL-MAP IE of the DL-MAP [27].

How often UL-MAPs are sent depends on the channel configuration and how the BS schedules the traffic [29].

2.3 Proposed methods and solutions

This subchapter describes implemented methods and solutions for different MAC procedures.

2.3.1 Sub-MAP overhead reduction

As discussed in subchapter 2.1 and subchapter 2.2, the 802.16 MAP entries result in a significant amount of overhead. As a solution, WiMAX Forum recommends using compressed MAP [30], which reduces the MAP entry overhead. On the other hand, the 802.16 standard introduces sub-MAPs which are appended to the compressed MAP format.

The 802.16 specification defines the maximum number of sub-MAPs that can appear in a frame. The exact number, configuration, and MCS to encode a particular sub-MAP are left undefined. To optimize the burst allocation efficiency, different algorithms for dividing SSs and choosing an appropriate sub-MAP for data burst allocation can be used. For instance, SSs can be divided into different groups each assigned to a different sub-MAP based on their SINR. As a result, signaling overhead for sending DL-MAP and UL-MAP is significantly reduced. Efficient use of sub-MAPs requires efficient algorithms to minimize the signaling overhead in the 802.16 system.

To construct sub-MAPs, the following questions must be answered:

1. How many sub-MAPs should we build? Even though the specification defines three sub-MAPs as the maximum value, it is not always efficient to construct and allocate all of them. Since each sub-MAP message has a fixed overhead, sometimes a better performance can be achieved if only one or two sub-MAPs are constructed or even when none is constructed.
2. What IE elements should a sub-MAP include? The basic 802.16e system operates 11 different MCSs. Thus, a decision must be made about which MCS IEs are to be placed into a particular sub-MAP.
3. What MCS should be used to transmit a sub-MAP? Being a MAC level management message residing in a separate data burst, the sub-MAP itself can be encoded with any MCS. It is important to ensure that the sub-MAP has an appropriate MCS so that all the contained IEs are successfully delivered to target stations.

2.3.1.1 Brute-force algorithm

The simplest approach to find the optimal number of sub-MAPs and their configuration is to test all the possible combinations. For each combination, the resulting number of slots is calculated. That number serves as a criterion for choosing the optimal case. The total number of combinations the brute-force algorithm needs can be expressed by the following expression:

$$1 + \sum_{i=1}^{S} C_N^i, \tag{1}$$

where S stands for the maximum number of sub-MAPs, N is the total number of MCSs, and

$$C_n^m = \frac{m!}{m!(n-m)!}. \tag{2}$$

In (1) we first test a configuration with no sub-MAPs, then we test configurations with one sub-MAP, and so on.

According to (1), this is an $O(N!)$ problem, which can be quite expensive computationally. However, if we substitute parameters from the 802.16 system into (1), i.e., $S = 3$ and $N = 11$, then the total number of combinations is 232. If we exclude three MCSs, QPSK1/2 Rep6…QPSK1/2 Rep2, that are not likely to be used with sub-MAP, then the number of MCSs is reduced to 8. Consequently, the number of iterations is 92, which is already a feasible number.

2.3.1.2 Advanced algorithm

To alleviate the computational burden of the brute-force algorithm, an iterative solution that requires much less computational resources is proposed. The solution exploits a very simple fact that it makes sense to start a sub-MAP on a particular MCS if that MCS has the biggest number of IEs. Thus, each MCS can be assigned a weight that determines a preference for a sub-MAP message

$$w_k = \frac{\sum S_i^{\mathrm{IE}}}{S_k^{\mathrm{slot}}},\ \forall k = 1 \ldots N, \tag{3}$$

where S_i^{IE} is the ith IE element size measured in bytes and S_k^{slot} is the slot size that corresponds to the kth MCS, for which a weight is calculated. Since the weight determines only a preference for a particular MCS where a sub-MAP might be built, an iterative algorithm is needed, simplified form of which is shown in Figure 6. It comprises the following stages:

1. Calculate a weight value for each MCS. Based on the number of uplink and downlink IEs, i.e., the number of data bursts that are encoded with the given MCS, each MCS is assigned a weight value that determines its preference for a sub-MAP candidate.
2. The list is sorted based on the weight values in the descending order, such that

$$w_{k-1} > w_k, \forall k. \tag{4}$$

3. Iteratively, consider the kth MCS as a candidate for a sub-MAP. The algorithm takes the (next) MCS with the highest weight value, adds a sub-MAP, and estimates the total number of slots that a new sub-MAP configuration

needs. If it reduces the total signaling overhead, then the MCS is saved into a permanent sub-MAP configuration. Otherwise, the MCS is just skipped. To determine whether the (next) MCS with the highest weight value reduces the total signaling overhead, the algorithm compares the total number of slots needed to encode IEs with the previous sub-MAP configuration. If the overhead is smaller in terms of slots, then update the permanent sub-MAP configuration.

4. The algorithm works until the maximum number of sub-MAPs is constructed or until all the MCSs are processed.

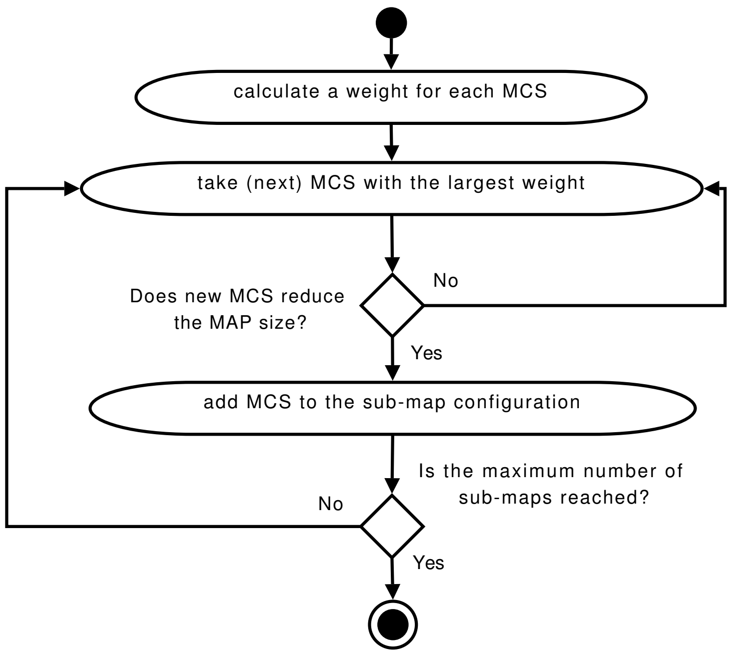


FIGURE 6 Sub-MAP construction algorithm.

From the implementation point of view, this algorithm does not require much processing resources. Its complexity can be estimated as $O(N)$ because in the worst case the algorithm will make as many iterations as the number of supported MCSs. However, in most cases the number of iterations corresponds to the maximum number of sub-MAPs, i.e., three iterations.

2.3.1.3 Comparison of algorithms

To analyze the performance of the proposed algorithm, the following criterion is used:

$$D = \frac{R - R^{\text{brute force}}}{R^{\text{brute force}}}, \tag{5}$$

where R is the total number of slots that a particular sub-MAP configuration needs. As considered in 2.3.1.1, the brute force algorithm always results in the best performance. Thus, this fact can be used as the efficiency criterion for any other algorithm. The generated 10^6 various configurations with different number of IEs for each MCS are passed to the brute-force and the advanced algorithm. For each configuration, the difference between two algorithms by using equation (5) is measured. Figure 7 presents the results in the form of a cumulative distribution function. As can be seen, in approximately 50% of cases both algorithms have the same performance in terms of required slots, i.e., D equals zero. The 95th percentile indicates that the difference is about 5%.

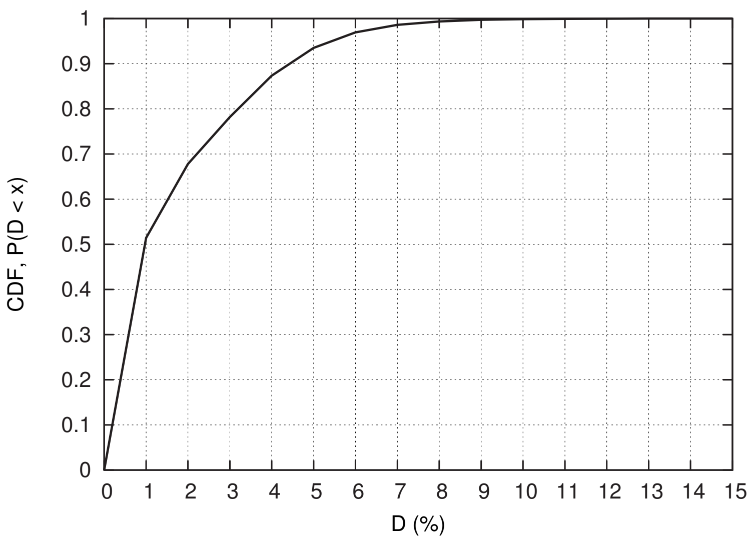


FIGURE 7 Algorithm performance.

2.3.2 MAP MCS

The 802.16 standard supports a variety of modulation and coding schemes, which can be changed on a burst by burst basis, depending on channel conditions [31],[32].

To ensure that the DL-MAP message is delivered with a high probability to all subscriber stations, it is possible to downgrade its MCS by one or several steps when compared to the most robust DL MCS over all the stations.

This feature is referred to as the *MAP link adaptation*.

[1] The same principle is applied for the sub-MAPs. The sub-MAP itself can be encoded with an MCS, which is zero or more steps more robust than the most robust data burst MCS it contains.

A sub-MAP may downgrade its MCS by at least one step. If a sub-MAP

1 It is worth mentioning that the MAP link adaptation is designed often in such a way that the MAP MCS never goes above some predefined MCS that determines the MAP coverage area, e.g., QPSK1/2 Rep2.

MCS is not downgraded at all, then there is quite a high probability that it will be lost at the SS side due to varying channel conditions.

At the same time, the number of bytes per slot should also be taken into account, as it impacts on the total transmitted data. The number of bytes per slot is varied, depending on the selected MCS, thus choosing an optimal MCS is very important for balancing between robustness and efficiency. Each MCS has a predefined capacity as shown in Table 1.

TABLE 1 Slot Capacity for various MCS values.

MCS	**DL Bytes per slot**	**UL bytes per slot**
QPSK 1/8	1.5	1.5
QPSK 1/4	3	3
QPSK 1/2	6	6
QPSK 3/4	9	9
QAM-16 1/2	12	12
QAM-16 2/3	16	16
QAM-16 3/4	18	16
QAM-64 1/2	18	16
QAM-64 2/3	24	16
QAM-64 3/4	27	not used
QAM-64 5/6	30	not used

2.3.3 IEEE 802.16 Persistent Allocations

In addition to previously discussed techniques, 802.16 adopts also Persistent Allocations as a mechanism to reduce signaling overhead by allocating resources persistently, thus eliminating the need to signal the allocation information in every frame [33], [34]. This is especially the case for VoIP applications, where the large MAP overhead is caused by dynamic scheduling of VoIP traffic [35]. Persistent allocations allow omission of signaling information over multiple frames and save considerable amount of MAP overhead.

However, if the MCS of some particular SS is changed due to link adaptation mechanism, then a BS needs to signal about the new allocation. Such a situation may bring inefficiency and results in resource wastage through inefficient resource allocation. As a solution to this problem, the SSs can be split into several groups based on their MCS [36].

3 OVERHEAD OF ARQ MECHANISM IN THE IEEE 802.16 SYSTEM

Providing VoIP services in the IEEE 802.16 BWA system with satisfying speech quality is difficult because of several factors, such as packet errors, delays and jitters. Effective error recovery mechanisms such as Automatic Repeat on reQuest (ARQ) are important as they can reduce packet loss. This chapter focuses on introducing efficient techniques to improve the performance of ARQ mechanism as well as choosing optimal ARQ related parameters.

3.1 IEEE 802.16 ARQ architecture

Wireless networks may discard packets due to a variety of reasons entirely unrelated to channel errors. These reasons include lack of queuing space, congestion management, faults, and route changes. Discarded packets cannot be used for further processing without a method of correction. For this purpose, the 802.16 standard [27] defines several techniques, among them ARQ. ARQ is used over a wide range of underlying physical media, including cellular wireless, wireless LANs, RF links, and other types of channel. ARQ is a mechanism by which the receiving end of a connection can request retransmission of MAC protocol data unit (PDU), generally as a result of having received it with errors. It is a part of the 802.16 MAC layer and can be enabled on a per-connection basis. When ARQ is enabled, the connections are set up and defined dynamically through the DSA/DSC class of messages.

The 802.16 ARQ mechanism is controlled by a number of parameters, such as the size of user data carried in a frame, the size of ARQ block, the size of PDU, and ARQ timers. The specification defines them, but it does not provide concrete values and solutions, and therefore it is left open for research. ARQ uses acknowledgements (messages sent by the receiver indicating that it has correctly received a packet) and timeouts (specified periods of time allowed to elapse before an acknowledgment is to be received) to achieve reliable data transmission over an unreliable service. If the sender does not receive an acknowledgment before the timeout, ARQ usually retransmits the packet until the sender receives an acknowledgment or exceeds a predefined number of retransmissions. Each acknowledgement/retransmission requires additional resources, and taking into account that ARQ is enabled at the connection, it is important to set up proper initial ARQ parameters. Otherwise, the enabled ARQ may result in a decreased QoS.

MAC PDU overhead is larger for non-ARQ connections than for ARQ enabled connections. First, each PDU must contain the CRC-32 field. Second, the PSH and the FSH sizes are larger since they carry the ARQ block sequence number (BSN). If a PDU carries a non-fragmented SDU, the fragmentation subheader (FSH) is still mandatory because it encodes the ARQ BSN. A typical ARQ connection should have a limited PDU size. A connection may experience a poor performance if large PDUs are constructed [37].

Next subchapters present the main ARQ-related features.

3.1.1 ARQ fragmentation/packing

ARQ mechanism assumes a segmentation of user data into blocks and their transmission to the receiver. For these purposes, the 802.16 standard defines several techniques such as fragmentation and packing. For ARQ enabled connections, fragments are formed for each transmission by concatenating sets of ARQ blocks with adjacent sequence numbers.

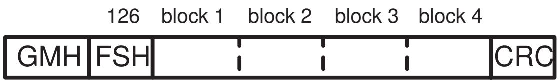
Enabling of fragmentation is optional [27], but it is encouraged, to improve link efficiency. When fragmentation is enabled, the transmitter partitions each SDU into fragments for separate transmission based on the value of the *ARQ_BLOCK_SIZE* parameter. A simplified form of fragmentation is shown in Figure 8(a). When fragmentation is not enabled, the connection is managed as if fragmentation was enabled. In this case, regardless of the negotiated block size, each fragment formed for transmission contains all the blocks of data associated with the parent SDU.

If packing is turned on for a connection, the MAC performs packing of multiple MAC SDUs into a single MAC PDU. A simplified form of packing is shown in Figure 8(b).

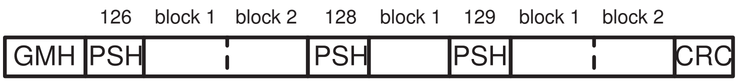
An SDU can be partitioned into multiple fragments that are then packed into the same PDU for their first transmission. At the same time, PDUs can have fragments from the same or different SDUs, including a mix of first transmissions and retransmissions.

3.1.2 ARQ block usage

A MAC SDU is logically partitioned into blocks whose length is specified by the ARQ_BLOCK_SIZE parameter. Once an SDU is partitioned into a set of blocks, that partitioning remains in effect until all blocks of the SDU are successfully delivered to the receiver, or the SDU is discarded by the transmitter state machine.



(a) fragmentation



(b) packing

FIGURE 8 ARQ blocks with fragmentation and packing mechanisms.

Sets of blocks selected for transmission or retransmission are encapsulated into a PDU. A PDU can contain blocks that are transmitted for the first time as well as those that are currently being retransmitted. If ARQ is enabled at the connection, FSH and PSH contain a BSN, which is the sequence number of the first ARQ block in the sequence of blocks following the subheader. It is up to a transmitter whether a set of blocks once transmitted as a single PDU should be retransmitted also as a single PDU. Figure 9 shows the rearranged PDU originally presented in Figure 8(b). The original PDU has been divided into two PDUs which contain the same three SDUs as the original PDU.

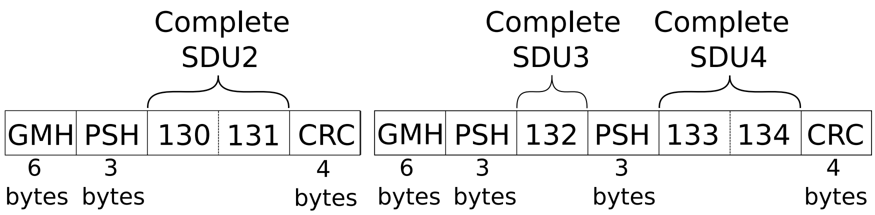


FIGURE 9 Rearranged PDU.

3.1.3 ARQ Feedback Types

A system supporting ARQ should be able to receive and process the ARQ feedback messages. The ARQ feedback information can be sent as a standalone MAC management message on the appropriate basic management connection or piggybacked on an existing connection. Piggybacked ARQ feedback is sent as follows: the ARQ feedback payload subheader can be used to send the ARQ ACK variants: cumulative, selective, selective with cumulative, and cumulative with block. When sent on an appropriate basic management connection, the ARQ feedback cannot be fragmented.

Cumulative feedback type uses the BSN value to indicate that its corresponding block and all blocks with lesser values within the transmission window have been successfully received. Selective feedback type defines each bit in the selective ACK MAP to indicate that the corresponding ARQ block has been received without errors. The bit corresponding to the BSN value in the ARQ Feedback Information IE is the most significant bit of the first map entry. Cumulative with selective feedback type associates cumulative and selective mechanisms: acknowledgement is made for a number of ARQ blocks.

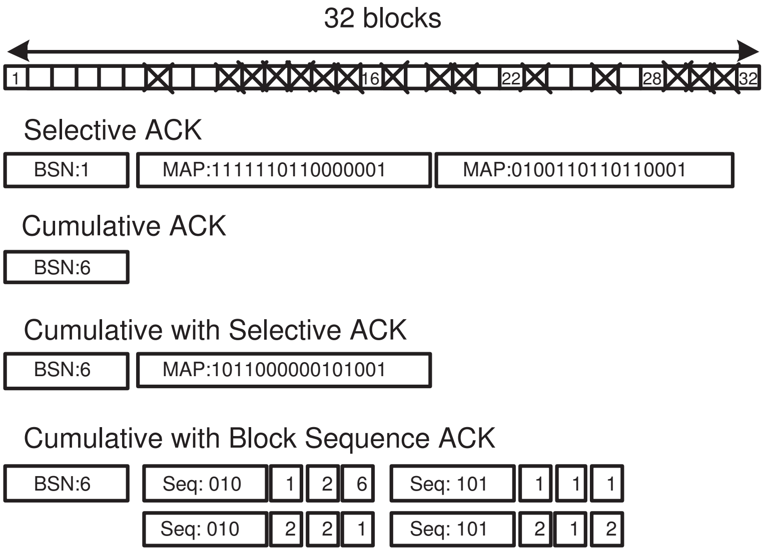


FIGURE 10 Example of ARQ feedback types.

Figure 10 presents an example in which every feedback type is applied to the same set of ARQ blocks. Selective ACK can acknowledge these 32 blocks in two maps. Cumulative ACK cannot acknowledge all the blocks because there are negative acknowledgements. Thus, only six blocks are encoded. Cumulative+selective ACK can send both positive and negative acknowledgements. However, since there should be 16 blocks per one selective map, some blocks remain unacknowledged. For this particular example, cumulative+sequence ACK can acknowledge only 28 blocks; one message can hold four sequence maps at most, whereas each map can have either two or three sequences. This type does not work effectively in this case because the block sequences are very short.

3.1.4 ARQ parameters

This subchapter describes some ARQ parameters. All the ARQ parameters listed below should be set when an ARQ enabled connection is set up.

– ARQ_BSN_MODULUS is equal to the number of unique BSN values.
– ARQ_WINDOW_SIZE defines the maximum number of ARQ blocks with consecutive BSN in the sliding window of ARQ blocks. The window is managed by the receiver and the sender. At any time, the sender may have a number of outstanding and awaiting acknowledgement ARQ blocks, which

are limited by this parameter. The next equation is applied to this parameter:

$$ARQ\_WINDOW\_SIZE \leq \frac{ARQ\_BSN\_MODULUS}{2}. \quad (6)$$

– ARQ_BLOCK_LIFETIME defines the maximum time interval an ARQ block can be managed by the transmitter ARQ state machine, once the initial transmission of the block has occurred. If a transmission (or subsequent retransmission) of the block is not acknowledged by the receiver before the time limit is reached, the block is discarded.
– ARQ_RETRY_TIMEOUT defines the minimum time interval a transmitter can wait before retransmission of an unacknowledged block for retransmission. The interval begins from the last transmission of an ARQ block.
– ARQ_SYNC_LOSS_TIMEOUT determines the maximum time interval the beginning of the ARQ transmit or receive window could be allowed to remain at the same value before declaring a loss of synchronization of the sender and receiver state machines when data transfer is known to be active.
– ARQ_RX_PURGE_TIMEOUT is the time interval the receiver can wait after successful reception of a block that does not result in advancement of the beginning of the ARQ transmission window. Logically, this timer works as the ARQ block lifetime at the receiver side.
– ARQ_BLOCK_SIZE defines the length used for partitioning an SDU into a sequence of ARQ blocks prior to transmission.

3.1.5 ARQ timers

The IEEE 802.16 specification defines several ARQ timers listed above. Figure 11 shows how they relate to the ARQ block states.

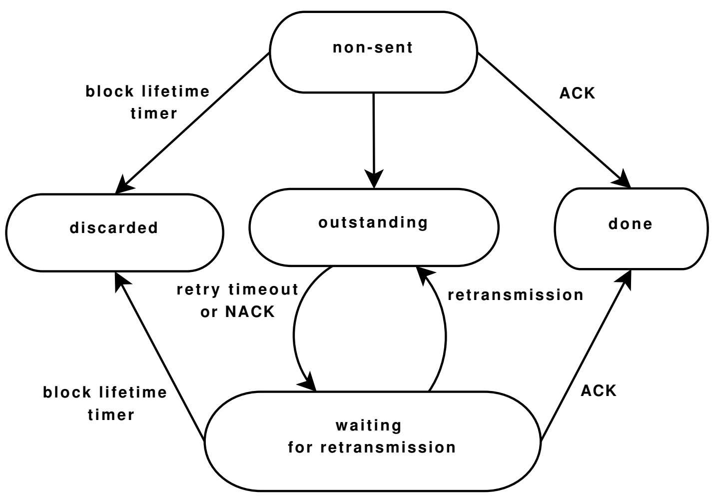


FIGURE 11 ARQ transmit block states.

An ARQ block may be in one of the following four states: not-sent, outstanding, discarded, and waiting- for-retransmission. First, as can be seen from Figure 11, any ARQ block begins as not-sent. Once it is sent, it becomes outstanding for a period of time termed ACK_RETRY_TIMEOUT, which determines the minimum time interval a transmitter can wait before retransmission of an unacknowledged block for retransmission. The interval begins from the last transmission of an ARQ block. While a block is in outstanding state, it is either acknowledged and discarded, or transferred to waiting-for-retransmission after ACK_RETRY_TIMEOUT or NACK. An ARQ block can become waiting-for-retransmission before the

ACK_RETRY_TIMEOUT period expires if it is negatively acknowledged. An ARQ block may also change from waiting-for-retransmission to discarded when an ACK message for it is received or after a timeout.

ARQ_BLOCK_LIFETIME determines the maximum time interval an ARQ block can be managed by the transmitter ARQ state machine once the initial transmission of the block has occurred. If a transmission (or subsequent retransmission) of the block is not acknowledged by the receiver before the time limit is reached, the block is discarded.

3.2 Proposed methods and solutions

This subchapter presents our solutions and proposals based on the topics discussed in previous subchapters.

3.2.1 A choice for the ARQ feedback type

The 802.16 standard defines ARQ feedback types, but does not provide an algorithm to choose an ARQ feedback type to achieve a good resource utilization.

Each feedback type has its advantages depending on the ARQ feedback transmission frequency, the error disturbance patterns, and computational complexity. From the implementation point of view, the selective feedback type does not require much processing resources because a connection simply puts information on the received blocks into the bitmap. On the other hand, a connection should try to rely upon the cumulative+sequence feedback type if resource utilization is of greater importance. However, it is more complex in implementation because block sequences must be detected. It could form an obstacle for a low power and low capacity mobile device.

Here we will propose an algorithm to choose an ARQ feedback type to achieve a good resource utilization. Our algorithm is based on the following assumptions: a) it is always more efficient to send positive acknowledgements by means of the cumulative type, and b) a sequence map can encode more blocks than a selective map. Indeed, the cumulative type can encode *any* number of ARQ blocks by using just one BSN number. Consequently, four sequence maps, each of which can have two sequences of 63 blocks, encode 504 blocks. If a map contains three short sequences, each of which can keep up to 15 blocks, then 180 blocks can be encoded. The proposed algorithm, simplified form of which is shown in Figure 12, comprises the following three stages:

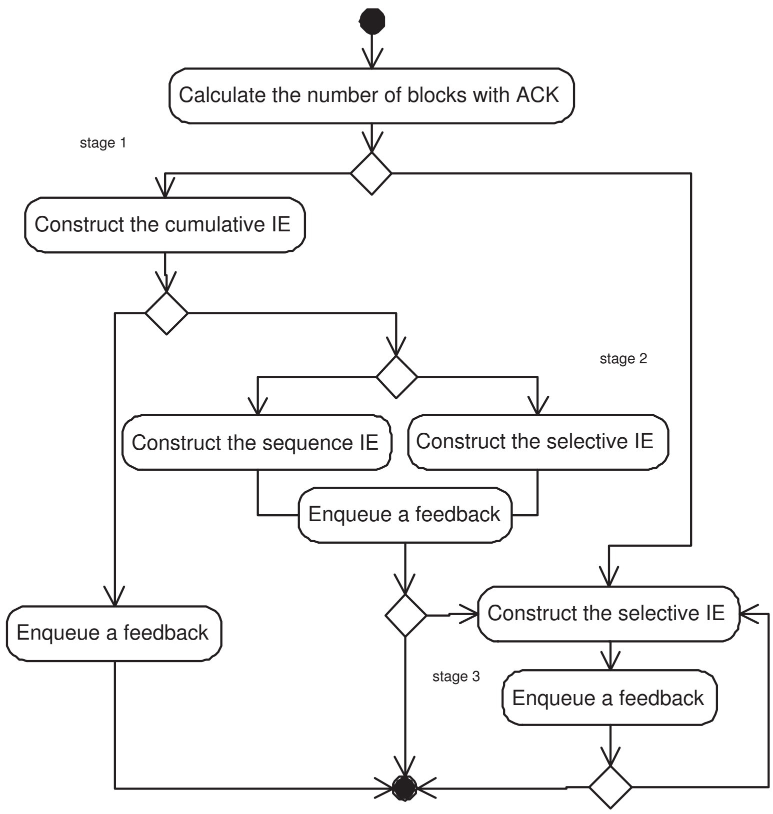


FIGURE 12 Algorithm to choose ARQ feedback types.

1. If there are positive acknowledgements in the beginning of the ARQ transmission window, construct the cumulative part. If there are no negative acknowledgements, then a single cumulative feedback message is created.
2. If there are remaining negative acknowledgements (optionally followed by positive and other negative acknowledgements), which we cannot send by using the cumulative part, then we have to choose a map type. To make a decision, we construct the sequence maps and calculate whether the selective maps can acknowledge more blocks. The maximum number of blocks to acknowledge selectively is 64, and it should be a multiple of 16. As for the sequence part, there is a limit for a sequence length and for the number of sequences we can send in one message. Eventually, we will have either the cumulative+sequence or cumulative+selective feedback type. As a choice is made, we "attach" map(s) to the cumulative part constructed at the previous stage.

3. Note that we can reach this stage in two cases. The first one is when there are no positive acknowledgements in the beginning of the ARQ transmission window and there is no way to create cumulative, cumulative+selective, or cumulative+sequence types. The second case to reach this stage is when neither cumulative+selective nor cumulative+sequence feedback types encode all the blocks. Although rare, it can happen, because both the cumulative+selective and cumulative+sequence types have technical limitations. It is important to note that we cannot create and send several consecutive cumulative+... feedbacks because the cumulative part of the second message will re-acknowledge positively those ARQ blocks that are acknowledged negatively in the first message. Regardless of the situation, we just create as many selective feedback types as necessary to acknowledge the remaining blocks. As mentioned above, four selective maps can acknowledge up to 64 blocks. It is important to note that due to the clarifications in [27], it is feasible to construct and send the cumulative+sequence feedback type when there are negative acknowledgements in the beginning of the ARQ window. It is possible to put out of the Tx window BSN field in the cumulative part so that it is ignored at the sender (receiver of the ARQ feedback). Such a solution eliminates the need for the selective type when there are errors in the beginning and improves the MAC overhead.

This algorithm scales well to the SS capabilities. If the selective type is not supported, then stage 3 will never be executed. If there is no support for one of the cumulative types, then stages 1 and 2 are simplified.

Referring back to stage 2, it is worth mentioning an algorithm to create sequences for the cumulative+sequence ARQ feedback type. The specification does not define it, thus allowing alternative implementations. As mentioned before, it is more complex in implementation because block sequences must be detected and correct sequence lengths must be constructed. To simplify this process, the algorithm uses two steps. On the first step, the algorithm parses all blocks and constructs sequences without checking any lengths. On the second step, the algorithm chooses sequence formats and, if necessary, splits large sequences into smaller ones so that they conform to the specification. The algorithm analyzes the current and the next sequence length to decide which sequence format should be used. As the sequence format is chosen, sequences are put into a map. If the sequence length exceeds the technical limit (63 for format 0 and 15 for format 1), then it is truncated and the remaining part is written into the input list so that it will be processed at the next iteration. The simplified form of this algorithm is shown in Figure 13. The algorithm stops when either all the sequences are processed or four maps are built. If there are not enough sequences to fill a single map, then zero lengths are put.

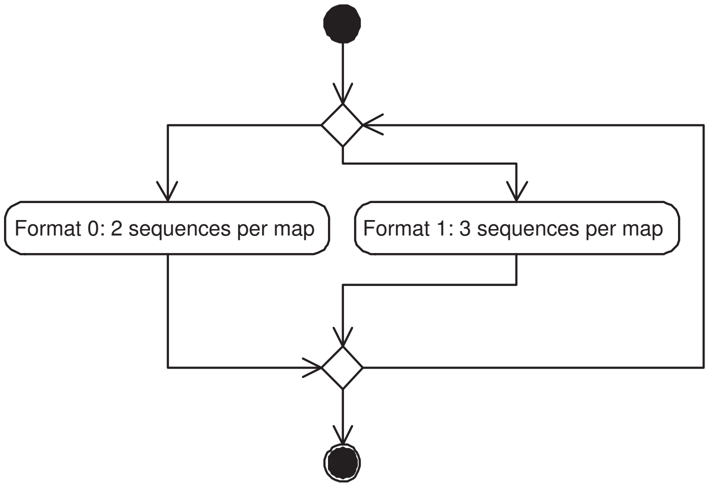


FIGURE 13 Algorithm to construct sequence maps.

As mentioned above, two sequence formats are available. The algorithm uses a simple, yet powerful, condition to select an appropriate sequence format:

$$\text{Format} = \begin{cases} 0, & (S_i > 15) \text{ OR } (S_{i+1} > 15), \\ 1, & \end{cases} \tag{7}$$

where S_i is the ith sequence length. The idea behind equation (7) is that it is more resource conserving to switch to sequence format 0 if either current or the next sequence is a large one. Otherwise, it is more efficient to use format 1 to encode more short sequences. As an example, Figure 14 presents the input sequences and constructed sequence maps (the first number is the sequence status and the number in the square brackets is a sequence length). As can be seen, the algorithm and the proposed conditions split the sequences efficiently into the maps according to the 802.16 specification.

0[65] 1[17] 0[3] 1[11] 0[4] 1[67] 0[40]

map #1	map #2	map #3	map #4
format 0	format 0	format 1	format 0
0[63] 0[2]	1[17] 0[3]	1[11] 0[4] 1[15]	1[52] 0[40]

FIGURE 14 Input sequence lengths and built maps.

To illustrate that the proposed algorithm selects efficiently the required sequence format, we present two cases for the same initial input sequences when only one particular format is in effect. As can be seen from Figure 15, both cases fail to encode all the blocks.

map #1	map #2	map #3	map #4
format 0	format 0	format 0	format 0
0[63] 0[2]	1[17] 0[3]	1[11] 0[4]	1[63] 1[4]

(a) format 0 maps only

map #1	map #2	map #3
format 1	format 1	format 1
0[15] 0[15] 0[15]	0[15] 0[5] 1[15]	1[2] 0[3] 1[11]

(b) format 1 maps only

FIGURE 15 Built maps (only one particular format).

The resulting computational complexity of the proposed algorithm to construct sequence maps is $O(2N)$. We need to make two passes: the first one is to calculate the initial sequence lengths and the second one is to split the sequences between the maps. The computational complexity of the selective map is $O(1)$.

4 SIMULATIONS

This chapter presents the dynamic simulation results in the 802.16 network. The main research tool for these studies has been the network simulator NS-2 [38]. NS-2 is a packet-level simulator [38]. When compared to dynamic system simulators, the packet-level simulator is very similar in terms of provided features. However, its protocol stacks and application behavior are modeled more accurately. Many PHY aspects are abstracted by means of simpler interfaces and models. Packet-level simulators allow simulating the access service networks because it is possible to define a network topology where base stations, routers, gateways, and clients send or receive data. This makes it possible to obtain true end-to-end simulation results. In general, NS-2 supports an array of popular network protocols, offering simulation results for wired and wireless networks alike. It can also be used as a limited-functionality network emulator. It is popular in academia for its extensibility due to its open source model and online documentation. NS-2 was chosen also because of the availability of the simulator kernel source code, a necessity for extending the simulation environment. The power of the NS-2 simulator is in the transport and application level. It is worth mentioning that it is unreasonable to get deep into PHY modeling as it will significantly increase the computational burden.

As any other simulator, NS-2 does have several disadvantages. NS-2 makes use of the flat earth model, in which it assumes that the environment is flat without any elevations or depressions. However the real world does have geographical features like valleys and mountains. Another disadvantage is that NS-2 simulations involve a laborious process of learning a scripting language, followed by having to understand in detail its inner working. For the researcher it takes some time to understand NS-2 completely.

It is to be noted that some existing simulation tools already support, to some degree, the IEEE 802.16 PHY and MAC levels. However, not all of them have all main MAC features.

To run simulations, WINSE extension was used with the 802.16 MAC and PHY levels implemented in the NS-2 simulator [26]. The MAC implementation contains all the main features of the IEEE 802.16 standard, such as downlink and uplink transmission, connections, MAC PDUs, packing and fragmentation, the contention and ranging periods, the MAC level management messages, and the ARQ mechanism. The HARQ implementation[26] supports Type I, i.e., chase combining (CC).

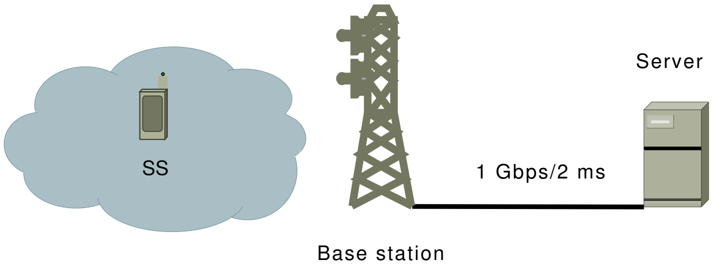


FIGURE 16 Network structure.

Figure 16 shows the network structure we used in the simulations. We concentrate on a single 802.16 sector with a BS and a number of SSs. The 802.16 network parameters are shown in Table 2.[1] The BS runs the throughput-fair scheduling algorithm, a simple, yet efficient, solution that is capable of allocating slots based on the QoS requirements, bandwidth request sizes, and the 802.16 network parameters. In a few words, it is based conceptually on the deficit round-robin; details of the algorithm are presented in [39], [29].

One of the ways to study the maximum achieved system gain is to use constant transmission. The VoIP application was chosen as the main application. Since it is a delay-sensitive application, it allows us to study the impact of the implemented methods on delays and packet drops. VoIP is also one of the main applications in NGMN networks.

Even though we study a symmetrical VoIP service, we choose the TDD DL/UL ratio in such a way that more resources are allocated to the DL sub-frame where signaling messages reside. While modeling the DL signaling messages, such as MAPs and sub-MAPs, we account for the fact that they can be dropped and apply the same error model as we do for normal data PDUs. As a result, if an SS loses a MAP or a sub-MAP message, it cannot receive or transmit data. This allows us to study how sub-MAPs impact on the performance of a VoIP service.

The BS runs the throughput-fair scheduling algorithm, general description of which is presented in [39].

Each SS establishes a basic management connection to exchange management messages with the BS. In addition, each SS has a bi-directional VoIP transmission with the server. That transmission is carried over two ertPS connections.

The VoIP application parameters given in Table 3 are taken from the IEEE 802.16m Evaluation Methodology Document [23]. However, to keep our focus on the system, we model the worst case scenario, where each VoIP SS has a constant active bi-directional VoIP transmission during each simulation run. In addition,

[1] These parameters conform to the WiMAX Forum mobile system profile [6].

TABLE 2 802.16 network parameters.

Parameter	Value
PHY	OFDMa
Bandwidth	10 MHz
Duplexing mode	TDD
Frames per second	200
Cyclic prefix length	1/8
TTG+RTG	296+168 PS
OFDM symbols	48
DL/UL symbols	28/19
DL/UL subcarrier alloc.	DL PUSC/UL PUSC
DL/UL slots	420/210
Channel report type / interval	CQICH / 40ms
Channel measurements DL/UL	preamble / data burst
Channel measurements filter	EWMA , $\alpha = 0.25$
Compressed MAPs	ON
sub-MAPs / max. number	ON / 3
Ranging transm. opport.	2
Ranging backoff start/end	1/15
Request transm. opport.	4
Request backoff start/end	2/15
CDMA codes	256
ranging+periodic ranging	64
bandwidth request	192
handover	–
HARQ	Type I (CC)
HARQ channels	16
HARQ buffer size	2048 B (per channel)
HARQ shared buffer	ON
HARQ max. retransmissions	4
HARQ ACK delay	1 frame
HARQ PDU SN	OFF
Fragmentation/packing	OFF

we limit the VoIP connection queue length to 3 SDUs to correspond to the air interface transmission delay of less than 100 ms. The ertPS connections use HARQ as a retransmission mechanism because ARQ cannot always ensure required delay limits. Furthermore, we turn the HARQ PDU sequence numbering off to cater for smaller delays.[2] In addition, we disable both packing and fragmentation for VoIP, which in conjunction with a proper VoIP scheduler ensures absence of unnecessary MAC level overhead. Uplink ertPS connections use the CDMA contention as a resumption mechanism to inform the BS that there is an uplink data to be transmitted. As simulated in [40], this method is more efficient than polling.

Using uncompressed headers results in significantly reduced performance,

2 The HARQ PDU ordering is usually turned on for the TCP connections where the TCP receiver may react unpredictably for packets arriving in the wrong order. In case of VoIP, a receiver usually implements a so-called de-jitter buffer, depth of which varies from 20 ms to 60 ms. An alternative VoIP configuration is to enable PDU SN and tune the HARQ ordering buffer timeout based on the maximum number of HARQ retransmissions.

TABLE 3 VoIP parameters.

Parameter	Value	
Codec	G.729	
Active state	100%	
Aggregation interval	20 ms	
Voice Payload	20 bytes	
802.16 Generic MAC header HARQ CRC	6 bytes 2 bytes	
IP/UDP/RTP headers	40 bytes (no PHS)	2 bytes (PHS)
Total VoIP packet size	68 Bytes	30 Bytes

TABLE 4 System level parameters.

Parameter	**Value**
Reuse factor	1/3
Inter-site distance	1.5 km
Path loss model	UMTS 30.30
Slow fading std.	8 dB
Fast fading	Ped B (60%), Veh A (40%)
Antenna technique	SISO (1x1)
Antenna pattern BS/MS	3GPP / Omnidirectional
Antenna height BS/MS	32 / 1.5 m
Tx power BS/MS	20 / 0.2 W

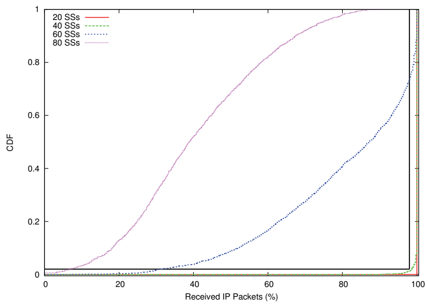
which would not be the case in practice. For this reason, Payload Header Suppression (PHS) was always enabled in these simulations as one of the ways to reduce upper layer overheads and to improve the number of supported users. PHS is a WiMAX feature. It allows the sender to not send fixed amount of headers and can reduce the 40 byte header overhead down to 3 bytes. Thus, PHS can significantly improve the capacity and should be used in any capacity planning or estimation.

To speed up simulations, no modeling of all the PHY details was done. We relied upon the effective SINR trace files taken from the system level simulator, where we had modeled 19 cells with 3 sectors per each cell. This provides quite a realistic fading in a wireless channel, causing packet errors and drops. The relevant parameters are given in Table 4. All the other PHY mechanisms, such as channel measurements, channel reporting, link adaptation, and scheduling are present in the simulation environment.

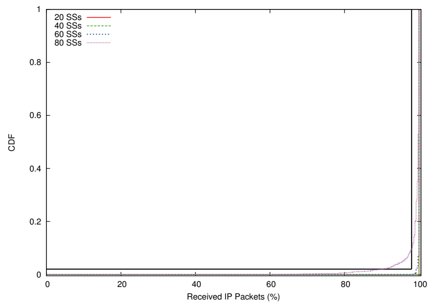
Extensive simulations were conducted to study signaling overhead of the whole system. For these purposes, the number of connections in the system was varied.

For each simulation case, many simulation runs were made to obtain proper confidence intervals. On average, each simulation run lasted for 13 seconds. Actual data transmission starts at the 3rd second because the first SSs have to enter the cell and register at the BS.

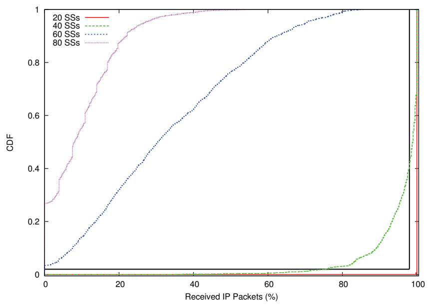
Figure 17 shows how injecting a different number of VoIP connections and enabling sub-MAPs impact the VoIP capacity and associated transmission delays. Boundary lines for X and Y - axes are introduced to see the cases where SSs do not meet their delay requirements. They show whether 98% of packets are delivered successfully to 98% of SSs within the delay bound of 100 ms.



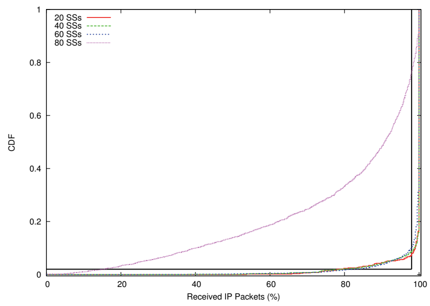
(a) disabled sub-MAPs, DL



(b) enabled sub-MAPs, DL



(c) disabled sub-MAPs, UL



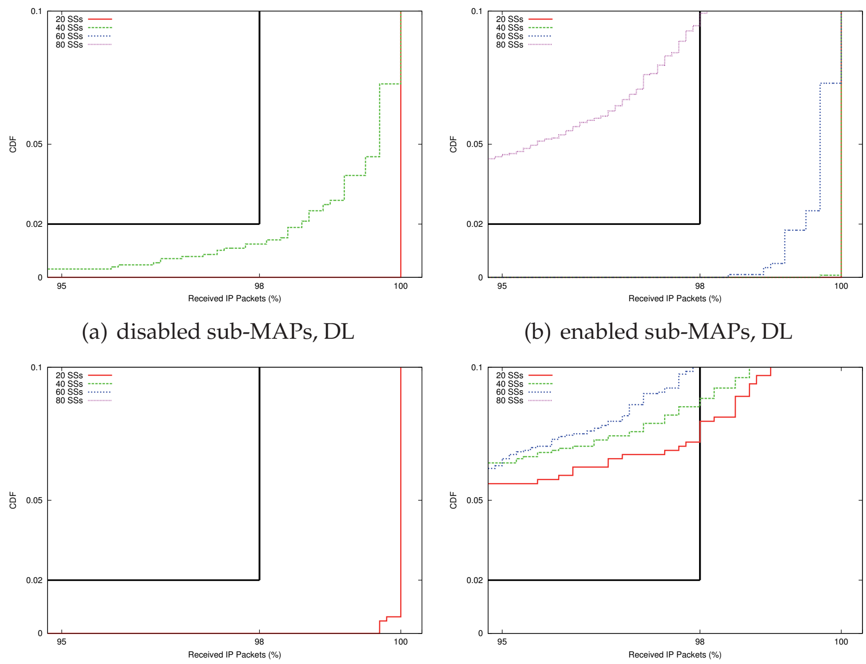
(d) enabled sub-MAPs, UL

FIGURE 17 VoIP SDU E2E delay CDF.

The results with end-to-end delay CDFs are shown in Figure 17(a) and Figure 17(c), for the downlink and uplink directions respectively. Figure 18(a) and Figure 18(c) demonstrate the same case with more elaboration for the downlink and uplink parts respectively. Figure 17(a) shows that if sub-MAPs are disabled then the downlink VoIP capacity is only 20 SSs. With a bigger number of SSs, downlink delays longer than 100 ms can be experienced. As can be seen from Figure 17(c), the uplink capacity is also limited to 20 SSs.

Next, we enabled sub-MAPs and analyzed the upper limits for the VoIP capacity. The results are shown in Figure 17(b) and Figure 17(d) for the downlink and uplink directions respectively. When compared to Figure 17(a), where the VoIP capacity is limited to 20 SSs in the downlink direction, Figure 17(b) illustrates clearly that enabled sub-MAPs improve overall VoIP capacity up to 60 VoIP connections. In the uplink, as can be seen in Figure 17(d), the VoIP capacity is around 60 SSs.

Figure 18(b) and Figure 18(d) show a more detailed elaboration for the down-



(a) disabled sub-MAPs, DL (b) enabled sub-MAPs, DL

(c) disabled sub-MAPs, UL (d) enabled sub-MAPs, UL

FIGURE 18 VoIP SDU E2E delay CDF. Detailed elaboration.

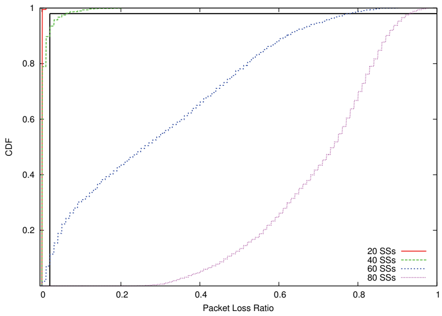
link and uplink parts respectively. By comparing Figure 18(a) with Figure 18(b), it can be seen that enabled sub-MAPs start to bring more gain.

Figure 18(d) also presents eloquently how enabled sub-MAPs tend to increase transmission delays for a small number VoIP connections. When compared to Figure 18(c), one can see that even 20 VoIP connections have slightly larger delays. The reason for this phenomena is dropped sub-MAP messages. If they are dropped, then both the DL-MAP and UL-MAP entries are lost. If an SS fails to receive the sub-MAP message, it cannot transmit anything in the UL direction, thus causing the queue to grow. It takes some time for the BS scheduler to detect this situation and allocate a larger UL grant.

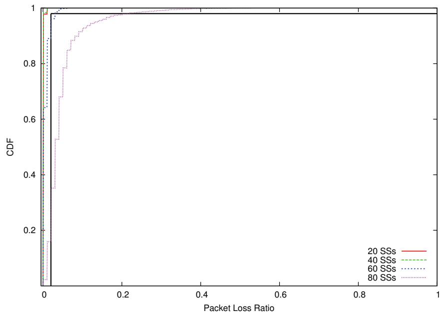
Figure 19 shows how different numbers of VoIP connections and sub-MAPs affect the VoIP capacity and the packet loss ratio. As in the case of delay analysis, we also apply boundary lines for X and Y - axes to ensure that 98% of SSs experience packet drops of less than 2%.

The results for packet loss ratio CDFs with disabled sub-MAPs are shown in Figure 19(a) and Figure 19(c) for the DL and UL parts, while Figure 20(a) and Figure 20(c) demonstrate a more detailed elaboration for the DL and UL parts respectively.

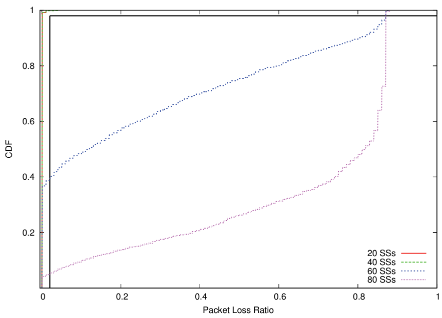
As seen from Figure 19(a) and Figure 19(c), the only case with 20 SSs shows a packet loss ratio of less than 2% for both the DL and UL directions. Due to frequent packet drops caused by queue overflows, all the other cases exceed the



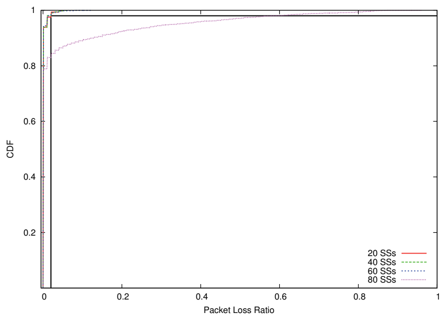
(a) disabled sub-MAPs, DL



(b) enabled sub-MAPs, DL

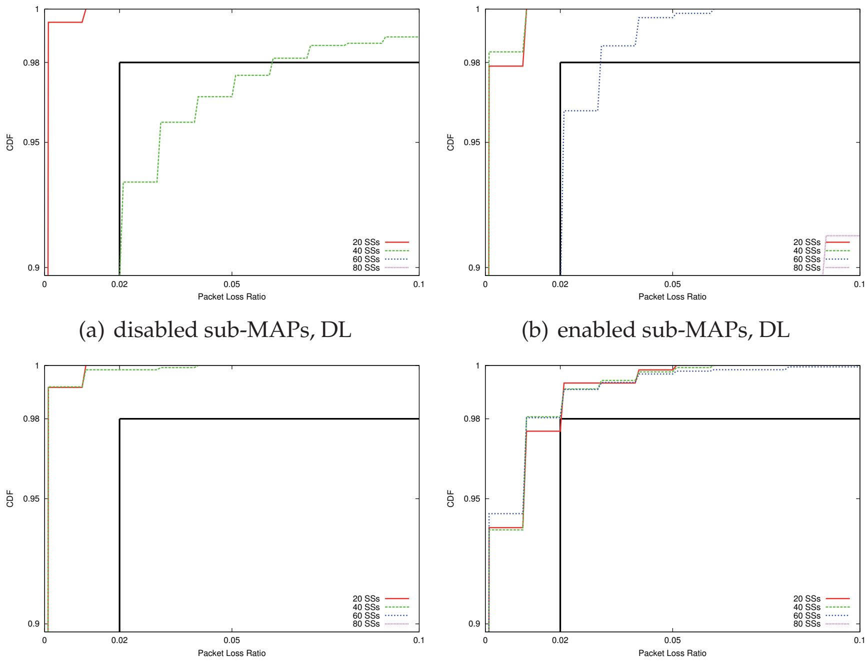


(c) disabled sub-MAPs, UL



(d) enabled sub-MAPs, UL

FIGURE 19 VoIP Packet Loss Ratio.



(a) disabled sub-MAPs, DL (b) enabled sub-MAPs, DL

(c) disabled sub-MAPs, UL (d) enabled sub-MAPs, UL

FIGURE 20 VoIP Packet Loss Ratio.

target packet loss ratio due to frequent packet.

As we enable sub-MAPs, we can drastically decrease VoIP packet drops because now the system has more resources to serve VoIP connections. As can be seen from Figure 19(b) and Figure 20(b), almost 60 DL VoIP connections can be served. As for the uplink, Figure 19(d) and Figure 20(d) show that enabled sub-MAPs decrease the packet loss ratio up to 60 SSs, but cannot meet the defined 2% with 80 SSs.

5 CONCLUSIONS AND FUTURE RESEARCH

The IEEE 802.16 family of standards has flexibility in network management, which means that different products that comply with the standard may have differences in performance and especially in the implementation of the layers beyond 802.16 MAC and PHY. While 802.16 would provide increased performance for users, the main factor for operators is capacity. With more users with smartphones and other devices, wireless bandwidth will continue to grow. Efficient solutions are needed in order to facilitate this phenomenon.

This dissertation has presented the most important solutions for balancing between network performance and complexity of implementation. In particular, the proposed solutions aim at mitigating the amount of overhead arising from the control signaling of the IEEE 802.16 system. It is also shown that the ARQ mechanism can improve significantly the performance of TCP based applications and should be enabled if the provider wants to ensure better QoS and to maximize the network utilization. The presented simulation results confirm that a correct configuration plays an important role in transmitting data over wireless channels in the IEEE 802.16 OFDMa system.

Further research will aim at studying more complex solutions where the sub-maps algorithm is applied to downlink and uplink completely independently, allowing even higher system performance. Even if bringing additional complexity, it has major advantages, which allow a more efficient use of the available resources. It will also be interesting to implement a persistent allocations mechanism to our NS-2 extension and perform extensive complex dynamic simulations when sub-MAP messages are used in conjunction with persistent allocations. It is expected that even more SSs can be served. Later, the obtained results can be compared with 3GPP LTE systems where a similar mechanism is available. It is the author's view that the proposed solutions for 802.16 are applicable to other broadband wireless access systems, such as 3GPP LTE.

YHTEENVETO (FINNISH SUMMARY)

Lisääntyvä älypuhelinten ja muiden langattomia verkkoa käyttävien laitteiden määrä aiheuttaa sen, että tarvitaan entistä tehokkaampia menetelmiä verkon suorituskyvyn parantamiseksi.

IEEE 802.16 -standardien määritykset ovat joustavia ja sallivat siten tuotteiden valmistajille erilaiset toteutustavat. IEEE 802.16 mahdollistaa laajakaistaisen langatoman verkkoyhteyden käyttäjille, ja verkko-operaattoreiden näkökulmasta yksi tärkeä asia on suurempi tarjottava oleva verkon kapasiteetti.

Tämä väitöskirja, jonka nimi on Hallintasignaloinnin aiheuttaman ylikuorman vähentäminen IEEE 802.16 OFDMa- järjestelmissä, esittää ratkaisuja verkon suoristuskyvyn parantamiseksi. Esitetyt ratkaisut keskittyvät erityisesti IEEE 802.16 -järjestelmän kontrolliviestien aiheuttamaan hukkasignalointiin. Väitöksessä osoitetaan myös, että ARQ- virheenkorjaus voi parantaa huomattavasti TCP-pohjaisten sovellusten suorituskykyä ja verkon käyttöastetta. Esitetyt simulaatiotulokset vahvistavat, että langattoman verkon oikeat parametrit ovat tärkeitä IEEE 802.16-järjestelmässä.

Työssä saatuja tuloksia voidaan jatkossa kehittää esimerkiksi submap-viestien avulla, jotka ovat määritelty täysin itsenäisesti lähetyksille ja vastaanotoille sallien vielä paremman järjestelmän suorituskyvyn. Myös pysyvän skeduloinnin toteutus WiNSE-moduuliin yhdistettynä submap-viesteihin on mielenkiintoinen jatkotutkimusaihe ja tämän avulla verkko voi siten palvella vielä useampia käyttäjiä.

Väitöskirjassa esitettyjä ratkaisuja voidaan soveltaa myös muihin langattomiin laajakaistaverkkoihin kuten 3GPP:n LTE -tekniikkaan.

REFERENCES

[1] J. Singer, “The WiMAX Forum Industry Research Report.” http://www.wimaxforum.org/resources/monthly-industry-report.

[2] S. Weber, X. Yang, G. de Veciana, and J. Andrews, “Transmission capacity of CDMA ad-hoc networks,” in *Spread Spectrum Techniques and Applications, 2004 IEEE Eighth International Symposium on*, pp. 245 – 249, Aug 2004.

[3] F. Adachi, D. Garg, S. Takaoka, and K. Takeda, “Broadband CDMA techniques,” *Wireless Communications, IEEE*, vol. 12, pp. 8 – 18, April 2005.

[4] R. Owen, S. Burley, P. Jones, and V. Messer, “Considerations for UMTS capacity and range estimation,” in *Capacity and Range Enhancement Techniques for the Third Generation Mobile Communications and Beyond (Ref. No. 2000/003), IEE Colloquium on*, pp. 5/1 –5/6, 2000.

[5] J. Chimeh, M. Hakkak, and S. Alavian, “Internet Traffic and Capacity Evaluation in UMTS Downlink,” in *Future Generation Communication and Networking (FGCN 2007)*, vol. 1, pp. 547 –552, Dec 2007.

[6] “WiMAX Forum Mobile System Profile Specification: Release 1.5 Approved Specification,” Aug 2009.

[7] 3GPP TS 23.228 v5.3.0, “LTE,” Jan 2002.

[8] C. Ball, T. Hindelang, I. Kambourov, and S. Eder, “Spectral efficiency assessment and radio performance comparison between LTE and WiMAX,” in *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pp. 1 –6, Sept. 2008.

[9] S. Srikanth and P. Pandian, “Orthogonal frequency division multiple access in WiMAX and LTE x2014; A comparison,” in *Communications (NCC), 2010 National Conference on*, pp. 1 –5, Jan. 2010.

[10] M. Chang, Z. Abichar, and C.-Y. Hsu, “WiMAX or LTE: Who will Lead the Broadband Mobile Internet?,” *IT Professional*, vol. 12, pp. 26 –32, May-June 2010.

[11] J. Chen, C.-C. Wang, F. C.-D. Tsai, C.-W. Chang, S.-S. Liu, J. Guo, W.-J. Lien, J.-H. Sum, and C.-H. Hung, “The design and implementation of WiMAX module for NS-2 simulator,” in *Workshop on NS-2*, Oct 2006.

[12] S. Kim and I. Yeom, “IEEE 802.16 simulator.” http://cnlab.kaist.ac.kr/802.16/ieee802.16.html.

[13] I. C. Msadaa, F. Filali, and F. Kamoun, “An 802.16 model for NS-2 simulator with an integrated QoS architecture,” in *Simutools*, Mar 2008.

[14] R. Rouil, “The network simulator NS-2 NIST add-on: IEEE 802.16 model (MAC+PHY),” tech. rep., Jun 2007.

[15] W. Furlong and R. Guha, “OFDMA Extension of NS-3 WiMAX Module,” in *Computer Modeling and Simulation (EMS), 2010 Fourth UKSim European Symposium on*, pp. 426 –431, 2010.

[16] J. Gross, H.-F. Geerdes, H. Karl, and A. Wolisz, “Performance analysis of dynamic OFDMA systems with inband signaling,” *Selected Areas in Communications, IEEE Journal on*, vol. 24, pp. 427–436, March 2006.

[17] X. Jin, J. Zhou, J. Hu, J. Shi, Y. Sun, and E. Dutkiewicz, “An Efficient Downlink Data Mapping Algorithm for IEEE802.16e OFDMA Systems,” in *GLOBECOM*, pp. 1–5, Dec 2008.

[18] J. Kim and D.-H. Cho, “Piggybacking Scheme of MAP IE for Minimizing MAC Overhead in the IEEE 802.16e OFDMA Systems,” in *Vehicular Technology Conference*, pp. 284–288, Sep 2007.

[19] H. S. Kim and S. Yang, “Tiny MAP: An efficient MAP in IEEE 802.16/WiMAX broadband wireless access systems,” in *ScienceDirect: Computer Communications*, vol. 30/9, pp. 2122–2128, June 2007.

[20] “Air interface for fixed broadband wireless access systems.” IEEE Standard 802.16, Jun 2004.

[21] J.-W. So, Maetan-dong, Yeongtong-gu, Suwon-si, and Gyeonggi-do, “An efficient uplink mapping scheme for VoIP services in the IEEE 802.16e OFDMA system,” in *AEU - International Journal of Electronics and Communications*, vol. 62, pp. 768–776, Nov 2008.

[22] J.-H. Yeom and Y.-H. Lee, “Efficient transmission of multicast maps in IEEE 802.16e.,” *IEICE Transactions*, vol. 91-B, no. 10, pp. 3157–3161, 2008.

[23] “IEEE 802.16m evaluation methodology document (EMD).” IEEE 802.16 Broadband Wireless Access Group, Jan 2009.

[24] Y.-R. Chuang, H.-W. Tseng, S.-T. Sheu, and C.-W. Su, “An Almost Overhead-Free Error Control Scheme for IEEE 802.16-Based Multi-hop Networks,” in *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pp. 1 –6, Dec 2009.

[25] Z. Becvar and R. Bestak, “Overhead of ARQ mechanism in IEEE 802.16 networks,” *Telecommunication Systems*, pp. 1–15, 2010. 10.1007/s11235-010-9296-0.

[26] A. Sayenko, O. Alanen, H. Martikainen, V. Tykhomyrov, O. Puchko, and T. Hämäläinen, “WINSE: WiMAX NS-2 Extension,” in *2nd International Conference on Simulations Tools and Techniques*, Mar 2009.

[27] "Air interface for fixed broadband wireless access systems - amendment for physical and medium access control layers for combined fixed and mobile operation in licensed bands." IEEE Standard 802.16e, Dec 2005.

[28] H. Martikainen, "Analysis of duplexing modes in the IEEE 802.16 wireless system," in *Wireless Conference (EW), 2010 European*, pp. 849 –856, april 2010.

[29] A. Sayenko, *Adaptive scheduling for the QoS supported networks*. PhD thesis, Jyväskylän yliopiston kirjasto/Julkaisuyksikkö, 2005.

[30] "WiMAX Forum System Evaluation Methodology," Jul 2008.

[31] *Fundamentals of WiMAX: Understanding Broadband Wireless Networking*. Prentice Hall, 2007.

[32] R. Cohen and G. Grebla, "Efficient allocation of CQI channels in broadband wireless networks," in *INFOCOM, 2011 Proceedings IEEE*, pp. 96 –100, april 2011.

[33] D. Jiang, H. Wang, E. Malkamaki, and E. Tuomaala, "Principle and Performance of Semi-Persistent Scheduling for VoIP in LTE System," in *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pp. 2861 –2864, sept. 2007.

[34] J. So, "Performance Analysis of a Semi-fixed Mapping Scheme for VoIP Services in Wireless OFDMA Systems," in *Wireless and Mobile Communications, 2009. ICWMC '09. Fifth International Conference on*, pp. 13 –17, aug. 2009.

[35] S. Shrivastava and R. Vannithamby, "Performance analysis of persistent scheduling for VoIP in WiMAX networks," in *Wireless and Microwave Technology Conference, 2009. WAMICON '09. IEEE 10th Annual*, pp. 1 –5, april 2009.

[36] S. McBeath, J. Smith, L. Chen, A. Soong, and H. Bi, "VoIP support using group resource allocation based on the UMB system," *Communications Magazine, IEEE*, vol. 46, pp. 114 –120, january 2008.

[37] H. Martikainen, A. Sayenko, O. Alanen, and V. Tykhomyrov, "Optimal MAC PDU size in IEEE 802.16," in *4th International Telecommunication Networking Workshop on QoS in Multiservice IP Networks*, pp. 66–71, Feb 2008.

[38] UCB/LBNL/VINT, "Network Simulator NS-2." http://www.isi.edu/nsnam/ns.

[39] A. Sayenko, O. Alanen, and T. Hämäläinen, "Scheduling solution for the IEEE 802.16 base station," *Computer Networks*, vol. 52, pp. 96–115, 2008.

[40] J. Lakkakorpi and A. Sayenko, "Uplink VoIP delays in IEEE 802.16e using different ertPS resumption mechanisms," in *Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pp. 157–162, Oct 2009.

ORIGINAL PAPERS

PI

PERFORMANCE ANALYSIS OF THE IEEE 802.16 ARQ MECHANISM

by

A. Sayenko, V. Tykhomyrov, H. Martikainen and O. Alanen 2007

The 10-th ACM International Symposium on Modeling, Analysis and
Simulation of Wireless and Mobile Systems

https://doi.org/10.1145/1298126.1298180

PII

PERFORMANCE EVALUATION OF THE IEEE 802.16 ARQ MECHANISM

by

V. Tykhomyrov, A. Sayenko, H. Martikainen, O. Alanen and T. Hämäläinen 2007

7th International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking

https://doi.org/10.1007/978-3-540-74833-5_13

PIII

ON ARQ FEEDBACK INTENSITY OF THE IEEE 802.16 ARQ MECHANISM

by

Vitaliy Tykhomyrov, Alexander Sayenko, Henrik Martikainen, Olli Alanen and Timo Hämäläinen 2008

International Conference on Telecommunications

https://doi.org/10.1109/ICTEL.2008.4652627

PIV

ANALYSIS AND PERFORMANCE EVALUATION OF THE IEEE 802.16 ARQ MECHANISM

by

Vitaliy Tykhomyrov, Alexander Sayenko, Henrik Martikainen and Olli Alanen
2008

Journal of Communications Software and Systems, vol 4, issue 1

http://dx.doi.org/10.24138/jcomss.v4i1.235

PV

LINK ADAPTATION THRESHOLDS FOR THE IEEE 802.16 BASE STATION

by

Oleksandr Puchko, Vitaliy Tykhomyrov and Henrik Martikainen 2008

The Second Workshop on NS-2

https://doi.org/10.4108/ICST.VALUETOOLS2008.4369

PVI

WINSE: WIMAX NS-2 EXTENSION

by

Alexander Sayenko, Olli Alanen, Henrik Martikainen, Vitaliy Tykhomyrov,
Oleksandr Puchko, Vesa Hytönen, Timo Hämäläinen 2010

Special Issue of Simulation: Software Tools, Techniques and Architectures for Computer Simulation

https://doi.org/10.1177/0037549710373334

PVII

DECREASING THE MAP OVERHEAD IN THE IEEE 802.16 OFDMA SYSTEM

by

V. Tykhomyrov and A. Sayenko and O. Puchko and T. Hämäläinen 2010

European Wireless Conference

https://doi.org/10.1109/EW.2010.5483393

PVIII

INCREASING THE VOIP CAPACITY THROUGH MAP OVERHEAD REDUCTION IN THE IEEE 802.16 OFDMA SYSTEMS

by

Vitaliy Tykhomyrov 2010

Mobile Lightweight Wireless Systems: Second International ICST Conference

https://doi.org/10.1007/978-3-642-16644-0_51