IMT Institute for Advanced Studies, Lucca

Lucca, Italy

A General Modular Framework for Resource Allocation in IEEE 802.16

PhD Program in Computer Science and Engineering

XX Cycle

By

Alessandro Erta

2008

The dissertation of Alessandro Erta is approved.

Program Coordinator: Prof. Ugo Montanari, University of Pisa

Supervisor: Prof. Luciano Lenzini, University of Pisa

Supervisor: Dr. Enrico Gregori, CNR

Tutor: Prof. Luciano Lenzini, University of Pisa

The dissertation of Alessandro Erta has been reviewed by:

Prof. Ian F. Akyildiz, Georgia Institute of Technology, Atlanta

Prof. Edward W. Knightly, Rice University, Houston

Prof. Bernhard Walke, University of Technology, Aachen

IMT Institute for Advanced Studies, Lucca

2008

To Barbara

Contents

List of Figures

List of Tables

Acknowledgements

This thesis is the result of joint work done with the University of Pisa. I could not express enough thanks to Luciano Lenzini and Enzo Mingozzi for their guidance during my PhD. I am especially indebted to Claudio Cicconetti and Andrea Bacioccola for their support and collaboration in doing my work. I am grateful to Leonardo Badia for his insightful advices and our intense discussions.

IMT Lucca mates also deserve thanks for putting up with me and entertaining me; Alessio Botta, Mike Zanda, Davide Bacciu as well as everyone in CSE and in the other PhD programs. I also would like to thank Silvia Lucchesi and the other PhD tutors for their patience and professionalism.

I can not leave out of this list the Fluidmesh crew who always supported my cause and appreciated the effort spent in this work.

Finally, thanks to my family for their encouragement and to my girlfriend Barbara Maffei for her unfailing support.

Preface

When I got my PhD, it was a time when there were just no jobs for PhDs. Period. PhDs were getting the lowest paid technician jobs, if they were lucky, in any kind of science.
— Shannon Lucid (1943 -)

A general modular framework to address the resource allocation problem in IEEE 802.16 is proposed in this work. Chapter 1 introduces the IEEE 802.16 standard in the context of the broadband wireless technologies and describes the resource allocation problem. The Medium Access Control (MAC) protocol and the physical layer of IEEE 802.16 are reviewed in Chapter 2. Our framework, called PIPER, is presented in Chapter 3. PIPER consists of three main sub-tasks, i.e., *grant scheduling*, *grant arrangement* and *grant allocation*. The grant scheduling sub-task can be abstracted from the IEEE 802.16 physical layer specification and it is thus described within Chapter 3. On the other hand, Chapter 4 and Chapter 5 address the grant arrangement and the grant allocation sub-tasks with respect to IEEE 802.16d and IEEE 802.16e, respectively. The Half-Duplex Allocation (HDA) algorithm, which can be employed by the Base Station as grant allocator in presence of Subscriber Stations with half-duplex capabilities in Frequency Duplexing Mode is also presented in Chapter 4. Sample Data Region Allocation (SDRA) is instead a heuristic algorithm for grant allocation in IEEE 802.16e with Orthogonal Frequency Division Multiple Access (OFDMA) as discussed in Chapter 5. Our solution will be extensively evaluated through simulation. Finally, conclusions are drawn in Chapter 6.

Abstract

The IEEE 802.16 MAC protocol is centralized, that is, a single Base Station (BS) controls the communication to and from a number of Subscriber Stations (SSs). The BS is responsible for guaranteeing the QoS requirements of each SS. However, the QoS objective is often in contrast with the system factors and constraints which can depend on the most heterogeneous system characteristics. These include the duplexing mode, the specific air interface used and, even, the SS hardware capabilities. In this work, the problem of meeting the QoS requirements of each SS while, at the same time, satisfying the overall system constraints will be referred to as *resource allocation*. Through a detailed analysis of the standard, we show that the latter is an overly complex task. To tackle the complexity, we thus propose to subdivide the overall resource allocation process into distinct, though sometimes strictly coupled, sub-problems. Hence, we developed a general modular framework where all the sub-problems can be solved by independent sub-tasks which can be combined together in a pipeline. We identified three main sub-tasks, i.e., *grant scheduling*, *grant arrangement* and *grant allocation*. We call our framework *PIPER* which recalls the pipelined organization. We will show that our approach greatly reduces the complexity of the resource allocation problem and can be successfully adopted in many different contexts. Within the PIPER framework we also propose two grant allocation algorithms, namely Half-Duplex Allocation (HDA) and Sample Data Region Allocation (SDRA) for IEEE 802.16 FDD with half-duplex SSs and IEEE 802.16e TDD for mobility support, respectively. We will evaluate the performance our solutions through extensive simulation.

Chapter 1

Introduction

They say a year in the Internet business is like a dog year..
equivalent to seven years in a regular person's life.
In other words, it's evolving fast and faster.
— Vinton Cerf (1943 -)

In 1998, Steve Jobs, Apple's CEO, declared to BusinessWeek that "a lot of times, people don't know what they want until you show it to them". In 2001, Apple released the first version of iPod, its award-winner mp3 player. iPod represents one of the most successful events in the high-tech consumer market. During the same period, the mobile phone industries were tripling their profits each trimester. In the last few years, the mobile telephone has become from a rare, expensive equipment of the business elite to a pervasive, low-cost personal item. In many countries, mobile telephones outnumber land-line telephones; in the U.S., 50 percent of children under fifteen have mobile telephones (kid06). Additionally, an even more subtle dissemination of wireless devices based on Wireless Local Area Network (WLAN) and Wireless Personal Area Network (WPAN) technologies has recently taken place. WLAN refers to wireless networks whose transmission range is in the order of tens (hundreds in open space) of meters, and is usually identified by the leading technology IEEE 802.11/WiFi. Most of the recent laptops are equipped with a built-in IEEE 802.11 card. On the other hand, WPAN wireless networks have a shorter range of few meters, and are thus typically employed to connect personal devices, such as headphones, scanners,

and printers, to a single laptop or desktop PC. Nowadays, no one can imagine his/her life without this large number of high-tech devices currently available on the market. People have been shown what they wanted.

Due to the high level of penetration of this technology in our every-day life, we are currently witnessing a global convergence of heterogenous services like voice, video and data communications into a common platform. Traditional Internet applications such as email, instant messaging, and Voice over IP are also becoming available on mobile devices like smartphones and palmtops. Therefore, the increasing demand of a fast connection towards the Internet available anytime-anywhere is one of the recent technological and research challenges. In fact, existing network solutions result inadequate to meet the requirements in terms of bandwidth, coverage and delay of the most recent applications. For instance, IEEE 802.11 hot-spots provide reasonably good connectivity. However, due to their limited transmission range, IEEE 802.11 networks are not suitable for wide area networks. 2.5G cellular services, such as the General Packet Radio Service (GPRS) or Enhanced Data rates for Global Evolution (EDGE), might represent a good alternative. However, they are not available everywhere and only provide a limited transmission rate. While this solution can be enough for sporadic email check, it is definitely inadequate to meet the requirements of advanced multimedia applications, such as Voice over IP or videoconference. Besides, 3G cellular technologies, such as UMTS HSDPA/HSUPA (HT06), which promise to increase the transmission rate and include support for multimedia applications, are still in the infancy phase. In 2001, a new IEEE standard, namely IEEE 802.16, was published with the goal of providing "last mile wireless broadband access as an alternative to wired broadband like cable and DSL" and, also, "fixed, nomadic, portable and, mobile wireless broadband connectivity" (For). At about the same time, the WiMAX forum (For) was formed as a non-profit organization to promote and certify the compatibility and interoperability of Broadband Wireless Access (BWA) products using the IEEE 802.16 specifications. At the beginning of 2008 the WiMAX forum included more than 522 members, which confirms the positive trend experienced by the IEEE 802.16 among equipment manufacturers and service providers.

Initially, IEEE 802.16 was aimed at providing high-speed Internet access in a Point-to-Multipoint (PMP) manner only. The support of Quality of Service

(QoS) was natively present since the first release, which clearly stated the role of IEEE 802.16 as a leading technology for the support to advanced multimedia applications. However, Line-of-Sight (LOS) was required, because the air interface of the 2001 release was based on Single Carrier (SC) at very high frequencies, i.e., above 11 GHz. This constraint severely affects the spreading of the technology, since it can significantly increase the setup cost of both Base Station (BS) and Subscriber Stations (SSs), e.g., because of the installment of roof-top antennae. Hence, during the subsequent years the standard has been amended so as to include support to non-LOS deployment, the final version being published in 2004 as IEEE 802.16d (80204).

The IEEE 802.16d release included three new air interfaces, tailored to different types of scenarios. First, the SCa interface is a modification of the original SC physical layer, with support to lower frequencies, i.e., below 10 GHz, and non-LOS. However, both research and industry soon abandoned this air interface, mostly because of its inefficiency in a urban environment. Second, the Orthogonal Frequency Division Multiplexing (OFDM) was specifically designed for fixed BWA, in both urban and rural scenarios. With this air interface, a *mesh* operating mode was also included in the MAC to possibly extend the network coverage. SSs operating in mesh mode act as relays of packets from other SSs forwarding them towards the BS or other SSs. Finally, the Orthogonal Frequency Division Multiple Access (OFDMA) was tailored to mobile BWA applications, even though support to Mobile Stations (MSs) was recently added in the 2005 release of the IEEE 802.16 standard, i.e., IEEE 802.16e (80205). Each air interface can be operated in both Time Division Duplexing (TDD) and Frequency Division Duplexing (FDD). In Table 1 we summarize the most important steps of the IEEE 802.16 standard evolution and the related features.

Regardless of the specific air interface, the IEEE 802.16 MAC protocol is connection-oriented and supports QoS at the connection level. The BS controls the communication to (downlink) and from (uplink) each SS in a centralized manner. An SS can have either full-duplex or half-duplex transmission capabilities. A full-duplex SS (FD-SS) can simultaneously listen to the downlink channel while transmitting data, whereas a half-duplex SS (HD-SS) cannot receive while transmitting, i.e., uplink/downlink transmissions from/to an HD-SS cannot overlap in time. We refer to this as the *half-duplex constraint*. Before transmitting to/re-

Date	Standard	Description
Dec. 2001	802.16	First standard release for LoS, PMP broadband wireless access application using 10-66 GHz spectrum.
Jan. 2003	802.16a	Amendment for NLOS operation using 2-11 GHz spectrum. PMP and mesh operation supported. OFDM is adopted.
Oct. 2004	802.16d	Completely revised version to replace the previous ones.
Dec. 2005	802.16e	Amendment to support mobility. OFDMA is adopted.

Table 1: IEEE 802.16 standard evolution and characteristics.

ceiving from the BS, any SS must request the admission of a new connection. If accepted, the BS is then responsible for meeting the requested QoS guarantees. The access to the medium is scheduled on a frame basis. At the beginning of each frame the BS must advertise the schedule of each uplink and downlink sub-frames to the SSs. In other words, each frame period, the BS explicitly informs the SSs about which portion of the uplink (downlink) sub-frame is reserved for their data transmissions (reception). This is done by defining the content of specific MAC messages, namely MAPs, which are broadcasted at the beginning of each downlink sub-frame. In other words, the MAPs messages are the final product of the overall process running at the BS which aims at solving the problem of meeting the QoS requirements of SSs while, at the same time, satisfying the system constraints. This process will be referred to as *resource allocation* in this work. The latter is a complex task which must take into account the following issues:

1. the BS has to ensure that admitted connections are provided with the negotiated QoS guarantees (CELM06). Therefore, the issue of determining which SSs are granted capacity in the next frame, and how many bytes they are allowed to transmit, must be addressed based on the QoS requirements of every connection;

2. signaling in IEEE 802.16 is *in-band*, i.e., the overall capacity of the downlink sub-frame is shared between control messages, which includes MAPs, and data messages. Therefore, all control messages must be considered as overhead from the user standpoint. Additionally, the size of the MAC control overhead strictly depends on the number of users currently served

within a frame, i.e., the higher the number of SSs served, the higher the overhead (BCEM07). When performing the resource allocation process, one desirable objective would be that of minimizing the MAC control overhead so as to maximize the capacity available to users;

3. the capacity available for data transmission in a frame cannot be granted arbitrarily, but must obey a number of constraints derived directly from the standard specification of the frame structure, which varies depending on the air interface adopted;

4. the BS must ensure the correct operation of HD-SSs, i.e., the half-duplex constraint must be verified at each frame.

As can be seen, the above issues are strictly coupled and often in contrast with each other. Designing an efficient algorithm able to solve the resource allocation problem by taking into account all these issues is challenging. More importantly, the above requirements vary depending on the specific system specification, e.g., the air interface adopted, the duplexing mode etc. Albeit a joint approach which aims at optimizing the overall resource allocation process in cross-layer manner is clearly desirable (BBT$^+$07), it cannot be evidently applicable in general to every situation because of the wide variety of the IEEE 802.16 system constraints and specifications described above. Solutions have been proposed with regard to specific issues. For example, a cross-layer approach for refining the existing handoff procedures specified in 802.16e MAC layer is proposed in (CH07). Furthermore, a sub-channel and power joint allocation method for multiple users in 802.16e OFDMA/TDD is investigated in (KC07). However, such optimizations usually increase the design complexity and, thus, extends the development time.

To address the resource allocation problem we thus propose a general framework which aims at decoupling any resource allocation issues from each other. The above issues are basically solved independently, i.e., forcing when necessary the breaking of mutual dependencies, and the corresponding solutions are then combined sequentially so as to provide the final allocation. We called our modular framework *PIPER*.[1]

[1]PIPER takes its name from the pipeline organization and from the legend of the Pied Piper of Hamelin who promised to the Hamelin townsmen a definitive solution for their problem with the rats asking for money in exchange. Despite his success, the people reneged on their promise and refused to

The rationale of PIPER relies on the observation that the resource allocation task in IEEE 802.16 is *complex*. The complexity is due to the fact that the resource allocation problem consists of several components which depends on each other. Etymologically speaking, the word *complex*, actually means "composed by parts". Therefore, we first de-compose the overall problem into primitive sub-problems and define their specific characteristics. The focus of our investigation then shifts from the general complex problem to *complicated* sub-problems which can be solved separately through independent sub-tasks. The common scientific analytic method can also be applied at this stage. We identified three main sub-tasks composing the resource allocation process, namely i) *grant scheduling*, ii) *grant arrangement* and iii) *grant allocation*, which are responsible for, respectively, i) guaranteeing QoS to admitted connection, ii) exploiting the standard specification so as to optimize resource allocation and, iii) coping with the constraints imposed by the MAC and physical layer of IEEE 802.16. We are aware that our approach might lead to a sub-optimal global solution with respect to an "ideal" joint approach. However, we will show that the PIPER framework undoubtedly exhibits a tremendous advantage from the design standpoint. In fact, PIPER provides great flexibility since not only can each sub-problem be solved separately but also different algorithms can be envisaged and evaluated to implement the same sub-task. Therefore, the development rate can be speed up through early prototyping. Additionally, one may even think to exploit cross-layer information when implementing those algorithms as our framework is general enough to permit it. In this respect, we will also show that the PIPER framework can be applied to very different contexts. For instance, we exploit PIPER for both IEEE 802.16d FDD with OFDM for fixed BWA and IEEE 802.16e TDD for mobility support where the OFDMA air interface poses additional constraints. Within the PIPER framework, we developed two algorithms, namely Half-Duplex Allocation (HDA) and Sample Data Region Allocation (SDRA), for grant allocation of HD-SSs in IEEE 802.16d with OFDM and allocation of data regions in IEEE 802.16e with OFDMA, respectively. We prove the former to be optimal in the sense that it is always able to find a feasible grant allocation provided that the grant arrangement sub-task respects a necessary condition. HDA has $\mathcal{O}(n)$ complexity where n is the number of grants. The latter is instead a heuristic algorithm

pay the rat-catcher. The Pied Piper took his revenge by kidnapping the children of Hamelin (Bro88).

which can be exploited also in presence of advanced error recovery techniques of the IEEE 802.16e physical layer such as Hybrid Automatic Retransmission ReQuest (H-ARQ). We extensively evaluate our solutions via simulation in many different scenarios with realistic configurations.

Chapter 2

IEEE 802.16

The engineer's first problem in any design
situation is to discover what the problem really is.
— Unknown

In this chapter we describe the basic IEEE 802.16 MAC and physical layer characteristics and we investigate the IEEE 802.16 specifications which are essential to define the resource allocation problem. As already pointed out, the IEEE 802.16 MAC supports several air interfaces. In this work, we focus on Orthogonal Frequency Division Multiplexing (OFDM) and Orthogonal Frequency Division Multiple Access (OFDMA) only as they are the target air interfaces for IEEE 802.16d and IEEE 802.16e, respectively. Since each air interface poses different constraints to the resource allocation problem, they will be discussed separately in Sec. 2.2.1 and Sec. 2.2.2, respectively.[1]

2.1 MAC Protocol

In IEEE 802.16 uplink (from SS to BS) and downlink (from BS to SS) data transmissions occur in separate time frames. In the downlink sub-frame the BS transmits a burst of MAC Protocol Data Units (PDUs). Since the transmission

[1] When discussing the MAC characteristics, for the ease of comprehension, we do not distinguish between Subscriber Stations (SSs) and Mobile Stations (MSs). Unless otherwise specified, we thus adopt the former notation.

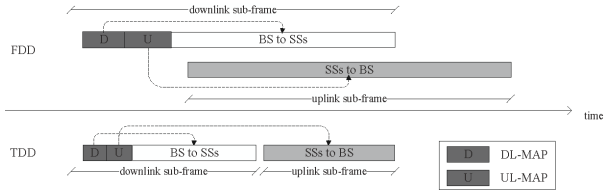

Figure 1: MAC frame duplexing with FDD and TDD.

is broadcast all SSs listen to the data transmitted by the BS. However, an SS is only required to process PDUs that are addressed to itself or that are explicitly intended for all the SSs. In the uplink sub-frame, on the other hand, any SS transmits a burst of MAC PDUs to the BS in a Time Division Multiple Access (TDMA) manner. Downlink and uplink sub-frames are duplexed using one of the following techniques, as shown in Fig. 1: Frequency Division Duplex (FDD), where downlink and uplink sub-frames occur simultaneously on separate frequencies, and Time Division Duplex (TDD), where downlink and uplink sub-frames occur at different times and usually share the same frequency.

The MAC protocol is connection-oriented: all data communications, for both transport and control, are in the context of a unidirectional connection that is uniquely identified through a 16-bit connection identifier (CID). The latter is included in the standard 6-byte MAC header that is appended to each PDU so as to identify the connection to which the encapsulated SDU belongs. In order to reduce the MAC overhead or improve the transmission efficiency the BS and SSs are (optionally) allowed to fragment a MAC Service Data Unit (SDU) into multiple PDUs, or they can pack multiple SDUs into a single PDU. Furthermore, a downlink/uplink burst usually consists of the *concatenation* of many PDUs (or parts thereof). A hybrid analytical-simulation study of the impact on the performance of this feature of the MAC layer has been carried out by (Hoy05). Results showed that, if the use of fragmentation is enabled, the frame can be filled almost completely, which can significantly increase the frame utilization, based on the size of SDUs. These optional features have also been exploited in a cross-layer approach between the MAC and application layers, so as to optimize the performance of multimedia streaming (SCGI05).

The IEEE 802.16 standard explicitly supports Quality of Service (QoS) at the

connection level. Therefore, the BS schedules the uplink and downlink connections so as to meet their negotiated QoS requirements. At the start of each frame, the BS advertises the computed schedule by defining the uplink and downlink grants of each SS. We define *grant* as the amount of time in an uplink (downlink) sub-frame reserved for an SS to transmit (receive). Based on this definition, each frame is actually an ordered sequence of grants. Each SS learns the exact grant sequence by decoding specific MAC control messages, i.e., MAPs, which are broadcasted by the BS at the beginning of each downlink sub-frame as shown in Fig. 1. MAPs consist of DL-MAP and UL-MAP which contain the downlink and uplink grants timetables, respectively. To guarantee that the SSs have enough time to decode the UL-MAP and schedule their transmissions, the uplink sub-frame is delayed with respect to the downlink sub-frame by a fixed amount of time, called the *uplink allocation start time*. IEEE 802.16 specifies that this value must be at least as long as the maximum Round Trip Time delay (RTT), but no longer than the frame duration.

To define the UL- and the DL-MAP, the BS must consider the actual amount of bytes which are present at each connection queue. Note that, while the BS has perfect knowledge of the status of the local downlink connection queues, it is not aware of the amount of data that is waiting for transmission at each connection in the uplink direction, i.e., the status of uplink connection queues which reside at SSs. Therefore, bandwidth in uplink is granted to SSs on demand. The IEEE 802.16 standard provides several mechanisms to inform the BS about the current backlog of each SS uplink connection. According to the standard, with *unsolicited granting* a fixed amount of bandwidth is requested on a periodic basis during the connection setup phase. After that, the requested bandwidth is granted automatically without any further explicit bandwidth request. On the other hand, with *demand assigned multiple access*, bandwidth is granted on a demand assignment basis, as the need arises. To do so, IEEE 802.16 provides SSs with dedicated control messages to notify the BS of bandwidth requests, and with a number of different mechanisms to convey such messages from the SSs to the BS. Bandwidth requests may be issued by either a 6-byte stand-alone Bandwidth Request (BR) PDU, or a 2-byte Grant Management (GM) sub-header, which is piggybacked on to a generic user data PDU. In both cases, the message specifies the amount of requested bandwidth, expressed as a number of user data bytes, and

it refers to a specific connection: in BR PDUs, the CID is explicitly specified, whereas, in GM sub-headers, it corresponds to the CID of the carrying PDU. Furthermore, requests may be incremental or aggregate. An *incremental request* indicates that additional bandwidth is needed, with respect to that requested so far by the same connection. On the other hand, an *aggregate request* resets any previous bandwidth request from the same connection. A request carried by a BR PDU may be either aggregate or incremental, whereas bandwidth requests piggybacked on GM sub-headers are always incremental.

Even though bandwidth requests are always per connection, the BS grants uplink capacity to each SS as a whole (80204) (par. 6.3.6.3), i.e., there is no explicit information in the UL-MAP message regarding the connections the grant was actually addressed to. Hence, the SS is not able to exactly predict to which connections the grant was intended. The SS MAC should thus implement a scheduling strategy to share the granted capacity to all (or some) of its connections (CELM06; CELM07).

Bandwidth requests themselves need bandwidth to be transmitted. A *unicast poll* consists of an uplink grant intended by the BS for a specific connection to transmit a bandwidth request. The grant is issued to the SS the connection belongs to, which will eventually schedule a BR PDU for that specific connection. On the other hand, *broadcast polls* are issued by the BS to all uplink connections, which contend for their use in a random access manner. We refer to a bandwidth request (carried by a BR PDU) sent in response to a broadcast poll from the BS as a *contention bandwidth request*.

Since it would not be feasible to address the QoS requirements of all of the applications foreseen for an IEEE 802.16 network, their functionality are grouped by the standard into a small number of classes named *scheduling services* based on the commonality of their: i) QoS service requirements (e.g., real time applications with stringent delay requirements, best effort applications with minimum guaranteed bandwidth); ii) packet arrival pattern (fixed/variable-size data packets at periodic/aperiodic intervals); and iii) mechanisms to send bandwidth requests to the BS. Thus, each scheduling service is tailored to support a specific class of applications. In the following, we describe the IEEE 802.16 scheduling services by focusing on the supported targeted applications, the related bandwidth request mechanisms and their specific MAC QoS parameters.

The IEEE 802.16 standard explicitly defines five scheduling services: Unsolicited Grant Service (UGS), real-time Polling Service (rtPS), extended real-time Polling Service (ertPS), non-real-time Polling Service (nrtPS), and Best Effort (BE).

UGS is designed for supporting real-time applications with strict delay requirements, that generate fixed-size data packets at periodic intervals, such as Voice over IP (VoIP) without silence suppression. The guaranteed service, is defined so as to closely follow the packet arrival pattern, hence grants occur on a periodic basis. With regard to uplink connections, capacity is granted by the BS regardless of the current backlog estimation, thus SSs never send bandwidth requests. In other words, grants are assigned with the same pace as that of packet generation. A drawback of UGS with uplink connections is that grants are assigned regardless of the actual backlog at SSs. For instance, UGS is not suitable for VoIP applications with silence suppression. In fact, while packets of fixed size are usually generated at a constant rate during talk-spurt (ON) periods, no packets are produced during silence (OFF) periods. Several modifications to the UGS have been proposed in the literature so as to efficiently support this kind of applications by reducing the bandwidth wasted for unnecessarily assigning uplink grants to idle VoIP connections. For instance, Hong *et al.* (HK06) proposed two alternative strategies. The first approach consists of dynamically adapting the grant interval, depending on the bandwidth that is actually consumed by the connection. The second is inspired to the UGS with Activity Detection (UGS/AD) scheduling service of the DOCSIS standard (CSR05), whose MAC-layer design choices have been broadly reused by the IEEE 802.16 working group. This strategy relies on the SS notifying the BS when it is entering/leaving an OFF period, respectively. In this way, during the OFF periods, the BS will assign small uplink grants instead of full-sized ones, which are used by the SS to notify the BS of the start of the next ON period as soon as the application becomes busy again. This solution has become part of the specifications of the IEEE 802.16e amendment (80205) as the Enhanced Real-time Polling Service (ertPS). Both these approaches have been shown to perform better than the original UGS, in terms of bandwidth utilization.

The rtPS scheduling service is designed to support real-time applications with less stringent delay requirements than UGS, that generate variable-size data pack-

ets at periodic intervals, such as Moving Pictures Expert Group (MPEG) video and VoIP with silence suppression. The key QoS parameters with such connections are the Minimum Reserved Traffic Rate and the Maximum Latency. The former specifies the minimum rate, in b/s, that must be reserved by the BS for this service flow whereas the latter, in s, upper bounds the interval between the time when an SDU is received by the MAC layer and the time when it is delivered to the physical layer. Since the size of packets with rtPS is not fixed as with UGS-tailored applications, SSs are required to notify the BS of their current bandwidth requirements. However, in order to grant deterministic access to the medium, the BS periodically sends unicast polls to rtPS connections. For this reason, the latter are refrained from using bandwidth request mechanisms on a contention basis. The polling period is equal to the Unsolicited Polling Interval (UPI), if specified.

With regard to uplink connections, the BS can issue periodic polls by means of a static timetable like that of UGS grants. The only difference is that the grant size is *not* equal to the expected SDU size, which is not known in advance, but to the number of bytes needed by an SS to transmit a bandwidth request PDU for the polled connection. In general, the bandwidth request mechanism of rtPS connections thus incurs an additional delay with respect to UGS connections of at least one frame duration. In fact, by responding to a unicast poll an SS notifies the BS of the backlog of one of its connections, but it cannot actually transfer data until the BS reserves an uplink grant for it. However, unlike UGS, rtPS connections can piggyback bandwidth requests on outgoing PDUs, provided that new data arrived before the previous backlog has been entirely served. This way an rtPS connection can anticipate the backlog notification to the BS with respect to the next scheduled unicast poll, thus reducing the aforementioned delay.

Unlike UGS and rtPS scheduling services, nrtPS and BE are designed for applications that do not have any specific delay requirements. The only difference between them is that nrtPS connections are reserved a minimum amount of bandwidth, which can boost performance of bandwidth-intensive applications, such as File Transfer Protocol (FTP) and Video on Demand (VoD). Both nrtPS and BE uplink connections typically use contention bandwidth requests; however, the BS should also grant unicast bandwidth request opportunities to nrtPS connections on a large time-scale, so as to enforce a minimum rate even at high network loads. In fact, in these conditions the BS could refrain from assigning

Bandwidth request mechanism	UGS	ertPS	rtPS	nrtPS	BE
Unsolicited granting	✓	✓			
Unicast polling		✓	✓	✓	
Broadcast/Multicast polling				✓	✓
Piggybacking			✓	✓	✓

Table 2: Bandwidth request mechanisms mapping to scheduling services.

slots for broadcast polls to save bandwidth.

When a BE or nrtPS uplink connection becomes busy after it has been idle for a long period, it notifies the BS of its backlog by responding to a multicast/broadcast poll. Therefore, the BS should reserve some capacity as multicast/broadcast polls so as to avoid starvation of idle BE connections. Recall that starvation of nrtPS connections is prevented by the large time-scale unicast polls issued by the BS. For instance, the BS can advertise the uplink capacity that is not scheduled as uplink grants available for broadcast polls. Also, reserving a small number of broadcast polls in each frame has been shown to improve the performance of BE connections, by reducing delays (CELM07). This is especially suitable for elastic applications, such as Web or FTP, which generate bursts of SDUs at irregular time intervals.

In Table 2, we report a summary of the the bandwidth request mechanisms used to support each scheduling service as recommended by the IEEE 802.16 standard.

2.2 Physical layer

The IEEE 802.16 standard includes several non-interoperable physical layer specifications. In this works, we specifically focus on the Orthogonal Frequency Division Multiplexing (OFDM) and the Orthogonal Frequency Division Multiple Access (OFDMA) air interfaces. The former has been envisaged by the WiMAX forum as the target air interface for fixed BWA, whereas the latter is the most suitable one for supporting mobility as specified by the IEEE 802.16e standard. Both OFDM and OFDMA are multiplexing techniques for Non-Line-of-Sight (NLOS) operations where the overall bandwidth is subdivided into multiple frequency sub-carriers (Cim85; NP00). Even though OFDM and OFDMA exhibit

several common aspects, for the sake of explanation, we describe them separately. This choice is motivated by the fact that the resource allocation process tremendously depends on the specific air interface adopted as described below.

2.2.1 Ortogonal Frequency Division Multiplexing

Figure 2 reports the OFDM frame structure in TDD. Each sub-frame consists of a fixed number of OFDM symbols. An OFDM symbol is made up from subcarriers, the number of which determines the Fast Fourier Transform (FFT) size used. The standard specifies an FFT size of 256. Part of the OFDM symbol duration, named the Cyclic Prefix duration, is used to collect multipath. The interested reader can find a technical introduction to the OFDM system of IEEE 802.16 in (KR04). The downlink sub-frame begins with a well-known sequence, called *long preamble*, to synchronize the receivers of the SSs. The duration of a long preamble is two OFDM symbols. A synchronization preamble, with duration of one OFDM symbol, namely *short preamble*, is also pre-pended to each downlink and uplink grant, so as to synchronize the BS and the SSs transceivers, before any data transmission. Immediately after the long preamble, the BS transmits the Frame Control Header (FCH), which consists of one OFDM symbol, and it is used by the SSs to decode the forthcoming MAPs messages transmitted by the BS. To notify a downlink (uplink) grant, a downlink (uplink) information element (IE) is included into the MAPs. The IE specifies the start and the end times of each grant and indicates the SSs to which the grant is addressed. As already pointed out, even though resources are scheduled by the BS on a per connection basis, grants are advertised per SS, i.e., the connection CIDs are not specified in the IE. Thus, SSs have to implement a scheduling function to select which of its connection will transmit in the assigned grant. Intuitively, the higher the number of IEs advertised by the MAPs, the greater the control message overhead, and, thus, the smaller the available frame capacity. Note that the standard defines several special IEs to manage particular conditions. For instance, Fig. 2 reports the Bandwidth Request IE which is used to notify the SSs that part of uplink sub-frame is reserved to broadcast polls. At the end of each downlink (uplink) sub-frame the BS provides the SSs with enough time to switch from reception (transmission) to transmission (reception). This time interval is referred

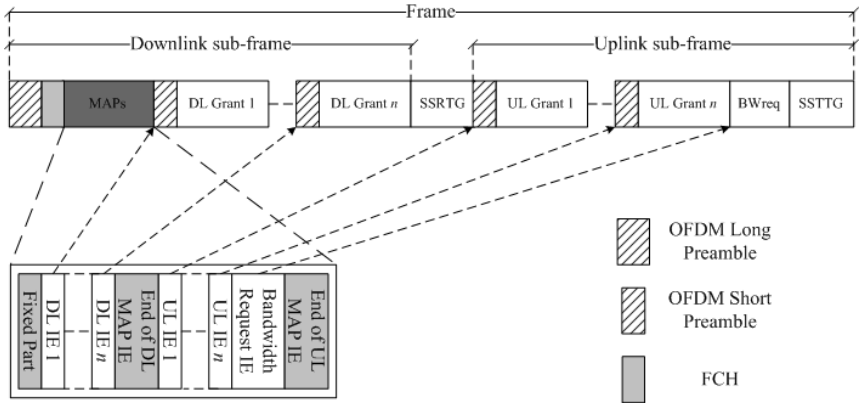

Figure 2: OFDM frame structure in TDD.

to as SS Receive to Transmit Gap (SSRTG), and the transmission to reception time is referred to as SS Transmit To receive Gap (SSTTG).

In order to exploit the location-dependent wireless channel characteristics, the IEEE 802.16 allows multiple burst profiles to coexist within the same network. In fact, SSs that are located near to the BS can employ a less robust modulation than those located far from the BS (Hoy05). The combination of parameters that describe the transmission properties, in downlink or uplink direction, is called a *Modulation and Coding Scheme* (MCS). Each MCS is associated with an Interval Usage Code (IUC), which is used as an identifier within the local scope of an IEEE 802.16 network. The set of MCSs that can be used is periodically advertised by the BS using specific management messages, i.e., Downlink Channel Descriptor (DCD) and Uplink Channel Descriptor (UCD). To maintain the quality of the radio frequency communication link between the BS and SSs, the wireless channel is continuously monitored to determine the optimal MCS. Specifically, the MCS is thus dynamically adjusted so as to employ the less robust profile such that the link quality does not drop below a given threshold, in terms of the Carrier-to-Interference-and-Noise Ratio (CINR) (EMSW02). However, as a side-effect of the dynamic tuning of the transmission rate, it is not possible for the stations to compute the transmission time of MAC PDUs *a priori*. Therefore, SSs always issue bandwidth requests in terms of bytes instead of time, without

including any overhead due to the MAC and physical layers.

Even though the link quality lies above a given threshold, it is still possible that some data get corrupted. To reduce the amount of data that the receiver is not able to successfully decode, several Forward Error Correction (FEC) techniques are specified, e.g., Reed-Solomon with Convolutional Code (RS-CC), which are employed in conjunction with data randomization, puncturing and interleaving. Data corruption of a MAC PDU can be detected by the receiver via an optional 32-bit CRC.

2.2.2 Ortogonal Frequency Division Multiple Access

The OFDMA air interface is specifically tailored to support mobility. In this context, we refer to terminals as Mobile Stations (MSs) instead of SSs. In OFDMA MAC frames extend over two dimensions: time in units of OFDMA symbols, and frequency, in units of logical sub-channels as shown in Fig. 3. A sub-channel is a subset of the available multiple frequency sub-carriers (Cim85; NP00). The standard specifies a few sub-carrier permutations, i.e., the mapping of logical sub-channels onto physical sub-carriers. Different mappings are tailored to different transmission environments and user characteristics. Two sub-carrier permutations are mandatory for downlink sub-frame: Partial Usage of Subchannels (PUSC) and Full Usage of Subchannels (FUSC); while PUSC is the only mandatory sub-carrier permutation for the uplink sub-frame. The FUSC method uses all the subchannels and employs fullchannel diversity by distributing the allocated subcarriers to subchannels using a permutation mechanism. On the other hand, in PUSC, subchannels are divided and assigned to three segments that can be allocated to sectors of the same cell. Subcarriers are distributed to subchannels through a different permutation mechanism. The interested reader can find further details in (Yag).

We define a zone as a portion of the frame in which one of the above subcarrier permutations is applied. Multiple zones within the same downlink or uplink sub-frame, employing different sub-channels mapping schemes, may exist as shown in Fig. 3 which depicts a sample OFDMA frame structure in TDD duplexing mode. The number of OFDMA symbols per frame depends on the frame duration, channel bandwidth, Fast Fourier Transform (FFT) size, and Cyclic Prefix

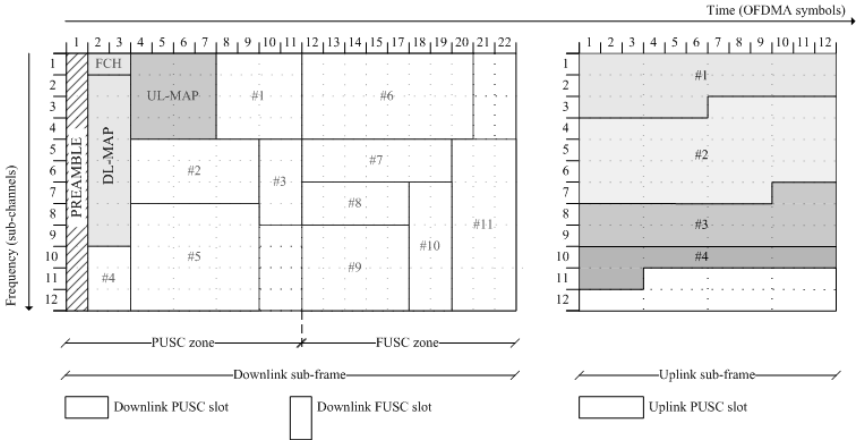

Figure 3: OFDMA frame structure in TDD.

length, whereas, the number of sub-channels depends on the channel bandwidth, FFT size, and sub-carrier permutation. The minimum resource allocation unit is the OFDMA *slot* (hereafter slot) and consists of one or more sub-channels by one or more OFDMA symbols, depending on the sub-carrier permutation. As for OFDM, the standard support different MCSs which can be used by the BS and MSs to adapt the transmission rate to the channel conditions. Therefore, while the dimensions of a slot are determined by the specific physical layer parameters and sub-carrier permutation, the amount of data which can be conveyed into a single slot varies depending on the robustness of the MCS, i.e., the more robust the MCS, the less the number of bytes carried by a single slot. The set of available MCSs, both for uplink and downlink transmission, is advertised by the BS on a regular basis through the DCD and the UCD control messages.

To better describe the OFDMA characteristics, without loss of generality, we exploit Fig. 3 as an example of a possible realistic system snapshot which shows a 5 ms frame with channel bandwidth of 10 MHz, 1024 FFT size, two downlink zones (PUSC and FUSC zones), and one uplink zone (PUSC zone). The structure of both downlink and uplink sub-frames is discussed below. With a slight abuse of notation, we can consider an OFDMA frame as a matrix where rows and columns are sub-channels and OFDMA symbols respectively (see Fig. 3).

First we describe the downlink direction. The downlink sub-frame begins with a physical preamble which is needed for physical synchronization. The preamble is one OFDMA symbol in the time domain, while it covers all the available sub-channels in the frequency domain. After the preamble, the BS transmits the FCH message. The DL-MAP message is thus allocated within the downlink sub-frame in column-wise order from the end of the FCH. If the DL-MAP size (in slots) exceeds the number of rows in a column, the allocation continues from the beginning of the next column. The remaining part of the downlink sub-frame is allocated by means of a number of *data regions*. A data region consists of a two-dimensional portion of the frame formed by a group of contiguous logical sub-channels in a group of contiguous OFDMA symbols. A data region may be visualized as a rectangle of OFDMA slots as shown in Fig. 3, and cannot span over two zones. However, MAC SDUs can be fragmented into many MAC PDUs, which in turn can be conveyed in different data regions.[2] If the size of the MAC PDUs of a data region is not a multiple of the number of bytes contained in the data region slots, the remaining portion is padded with stuff bytes. The size of each data region and its coordinates, with respect to the downlink sub-frame, depend on the resource allocation process and are specified by the BS in the DL-MAP message, which is advertised at the beginning of the downlink sub-frame in a PUSC zone. The standard allows for data directed to MSs employing the same MCS to be grouped into a single data region so as reduce the overhead. In fact, each data region is specified through an IE into the DL-MAP, which can optionally include the list of connections to which the data region is addressed, namely the *data region recipients*. Therefore, the greater the number of data regions, the higher the signaling overhead. With regard to the uplink direction, MSs access the medium in accordance with the uplink data regions advertised by the BS. Unlike downlink, uplink data regions are explicitly defined by the standard as a number of contiguous slots, starting from the upper-left corner of the uplink sub-frame, allocated in row-wise order as shown in Fig. 3. The end of each uplink data region is the beginning of the next one. As for downlink, each uplink data region incurs the overhead of one IE in the uplink map (UL-MAP) message. The

[2]This is only true provided that variable-length fragmentation of MAC SDUs into multiple MAC PDUs is allowed. While the latter is an optional feature of the IEEE 802.16 MAC, it can be effectively employed to reduce the MAC overhead due to padding the unused portion of data regions, and is thus specified as a mandatory feature by the WiMAX compliance datasheets.

latter is transmitted by the BS in the PUSC zone of the downlink sub-frame.

To increase the range and robustness of data transmission, the IEEE 802.16e standard specifies several advanced techniques, including Hybrid Automatic Retransmission ReQuest (H-ARQ). H-ARQ, first suggested in (WH61), uses an error control code in conjunction with the retransmission to ensure reliable transmission of data packets. In IEEE 802.16e, H-ARQ can be enabled on a per-connection basis for fast recovering from channel errors at an intermediate level between the MAC and physical layers (Yag). An H-ARQ data regions consists of a set of H-ARQ sub-bursts, each containing one or more MAC PDUs followed by a Cyclic Redundancy Check (CRC). The latter is used to determine whether the MAC PDUs have been received correctly. If this is not the case, then the H-ARQ sub-burst is re-transmitted, employing the same MCS. The H-ARQ receiver then combines the retransmissions along with the original transmission so as to infer the correct sequence of bytes transmitted. Note that this mechanism is substantially different from classical ARQ schemes, such as stop-and-wait, employed at the MAC layer of wireless MAC protocols (e.g., IEEE 802.11) where the MAC PDU is retransmitted until it can be fully recovered in its own, without combining the failed transmissions previously received. Furthermore, acknowledgments of the correct/incorrect reception of the H-ARQ sub-bursts are sent via a dedicated logical channel in the MAC frame, which ensures fast convergence of the H-ARQ process, and hence low transmission latencies. Preliminary works have shown that in the low SNR regime (below 4-5 dB), H-ARQ greatly increases the data rate (GWAC05).

As far as the resource allocation is concerned, the critical challenge with H-ARQ is that sub-bursts cannot span over two data regions. However, in the downlink direction, H-ARQ sub-bursts that are addressed to different MSs can be packed into a single downlink H-ARQ data region. Unlike non-H-ARQ downlink data regions, these MSs need not to employ the same MCS. In fact, each H-ARQ downlink data region is advertised in the DL-MAP by means of a specific IE, which specifies the starting point (top-left corner), width (number of OFDMA symbols), height (number of logical sub-channels), and number of H-ARQ sub-bursts. Each H-ARQ sub-burst is then advertised within the downlink H-ARQ data region IE by means of the length (in number of slots), the MCS, and the sub-burst recipient.

2.3 Half-duplex constraint

Regardless of the physical air interface adopted, the IEEE 802.16 standard provides support for both full-duplex and half-duplex SSs. A full-duplex SS (FD-SS) can simultaneously listen to the downlink channel while transmitting data, whereas a half-duplex SS (HD-SS) cannot receive while transmitting, i.e., uplink and downlink transmissions from/to an HD-SS cannot overlap in time. In fact, HD-SSs have only a single transceiver which can be set to a single frequency channel at any point in time. We refer to this as the *half-duplex constraint*. The latter is very stringent when the system is operating in FDD mode since the uplink and downlink channels occur simultaneously in time using separate frequency channels. Therefore, resource allocation must be performed so as to avoid uplink and downlink grants of HD-SSs to overlap in time. Despite this constraint, IEEE 802.16 FDD systems with HD-SSs are expected to be widely used solutions for a number of reasons. First of all, most licensed bands intended for data applications operate with FDD systems in mind (CLW$^+$06). Second, providing wireless communication devices with full-duplex capabilities is challenging (BEFN04). In fact, when the wireless transceiver is transmitting data, a large fraction of the signal energy leaks into the receive path. The transmitted and received power levels can differ by several orders of magnitude, and therefore the impact of the energy leakage can be significant. FD-SSs are thus more expensive to design and manufacture than HD-SSs, and the latter seem to offer a very attractive solution for user radio devices. Finally, up until the beginning of 2008, most WiMAX-certified SSs supported only half-duplex operations, when operated in FDD mode. Note that the half-duplex constraint only depends on the SSs hardware capabilities and it is thus a physical limitation which holds irrespective of air interface and the MAC protocol used.

Chapter 3

PIPER modular framework

And I chiefly use my charm
On creatures that do people harm,
The mole and toad and newt and viper;
And people call me the Pied Piper.
— Robert Browning (1812 - 1889)

In this chapter we present PIPER, a modular framework which can be adopted to address the resource allocation problem. As already discussed, in IEEE 802.16, the BS is in charge of allocating capacity in both downlink and uplink. This is accomplished by advertising capacity grants to SSs in both DL- and UL-MAPs on a frame-by-frame basis. However, neither a mandatory nor a reference algorithm is specified by the standard to perform such task, which is manufacturer-specific. On the other hand, the standard specifications, summarized in Chapter 2, impose a number of constraints and requirements on the candidate algorithm to be implemented, which makes its definition an extremely complex task. We identify three main issues, which are outlined in the following.

First, the BS has to ensure that admitted connections are provided with the negotiated QoS guarantees (CELM06; CELM07). Resource allocation can significantly impact the QoS of a given connection. Therefore, the issue of determining which SSs are granted capacity in the next frame, and how many bytes they are allowed to transmit, must be based on the QoS requirements of every connection.

Second, the capacity available for data transmission in a frame cannot be granted arbitrarily, but must obey a number of constraints derived straightforwardly from the standard specifications. For instance, the scheduled connections can be grouped together so as to form grants in several ways. Strategies can be envisaged to reduce the overhead required to advertise the grants. However, these strategies may be in contrast with the QoS scheduling of the connections.

Third, the scheduled grants must be allocated within the frame according to the frame structure. The latter depends on the specific air interface adopted. For instance, with the OFDMA air interface, downlink data regions must have rectangular shape and must be allocated in the sub-frame without overlapping with each other and without spanning over multiple zones. Additionally, proper system design must ensure that the half-duplex constraint is always satisfied for HD-SSs.

In order to tackle the complexity of the resource allocation problem, we propose a pipeline approach: each of the above issues is basically solved independently of each other, i.e., forcing when necessary the breaking of mutual dependencies, and the corresponding solutions are then combined in sequence in order to provide the final allocation. In particular, we envisage the frame allocation task as being split into three separate and sequential (though concurrent) sub-tasks, namely *grant scheduling*, *grant arrangement*, and *grant allocation*.

The grant scheduling sub-task takes as input the connections' QoS requirements and their current traffic load and determines how many MAC PDUs must be transmitted for each admitted connection so as to satisfy their requirements. Provided that a given MCS is associated to each SS, this turns into an amount of capacity, in slots, to be granted to each connection. The data region arrangement sub-task takes as input the output of the grant scheduling sub-task, and arranges the selected grants into different groups according to specific arrangement strategies, which aim at, e.g., minimizing the map overhead or providing better service in terms of packet delay to the SS. Such groups are then fed to the last sub-task, i.e., grant allocation, which is responsible for defining the final map contents, possibly re-arranging grants to fit them into the frame structure, and re-sizing scheduled grants to make the allocation feasible.

A schematic representation of PIPER is reported in Fig. 4. Grant scheduling selects a list of connections and schedules the amounts of bytes to be transmit-

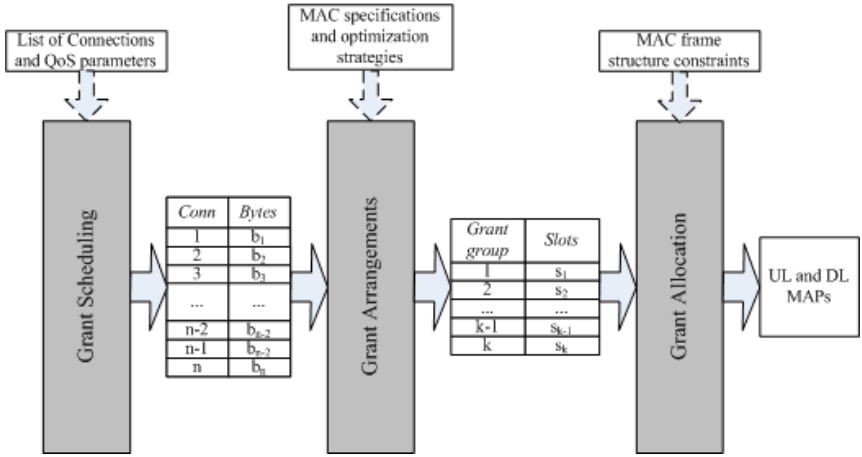

Figure 4: Piper modular framework organization overview.

ted (received) to (from) each connection based on the current connection queues status. This process will be described in Sec. 3.1. The data region arrangement module groups the list of connection grants into sets so as to exploit the proper MAC-specific features provided by the IEEE 802.16 standard. Finally, the grant allocation module is in charge of defining the final layout of the frame. In other words, the final output of the grant allocation module is the actual definition of the UL- and DL-MAPs. Since grant arrangement and grant allocation depend on the specific characteristics of the IEEE 802.16 physical layer, for the ease of comprehension, they will be discussed with respect to IEEE 802.16d and IEEE 802.16e in Chapter 4 and Chapter 5, respectively.

Note that, PIPER complies with an open-loop system design. In fact, we intentionally avoid any feedback between subsequent modules. The rationale of this choice is as follows. Since bandwidth is scheduled on a frame-by-frame basis, QoS guarantees should be expressed with a time granularity no smaller than the frame duration. In fact, several values are allowed by the standard for the latter — ranging from 2.5 to 20 ms — and the network operator can choose the one which best fits the service it is planning to offer. It follows that, with the pipeline approach in mind, QoS provisioning only depends on the algorithm implemented in the grant scheduler. This is the main advantage of this approach,

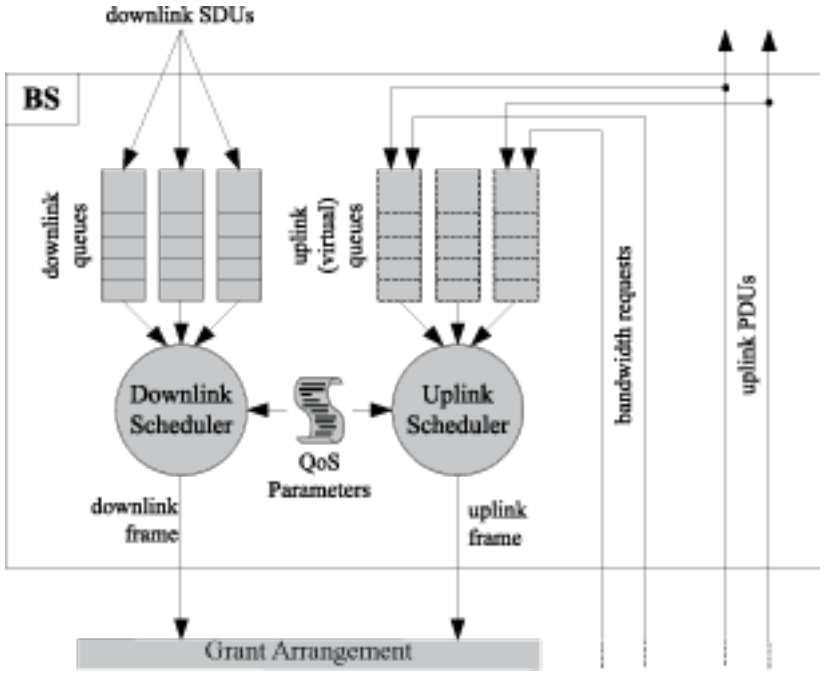

Figure 5: Grant scheduler architecture.

since the task of scheduling grants for QoS support is confined to a well-identified functional sub-module, and is naturally abstracted from the details originating from the grant allocation within the frame.

Furthermore, it is easy to see how such an isolation enables a wide variety of scheduling algorithms to be implemented in the grant scheduler sub-task. This happens as a result of the simple adaptation of well-known algorithms already proposed in the context of wired networks, where this discipline has been extensively studied in the recent past. Finally, the PIPER approach has a definite advantage as far as the implementation is concerned. In fact, MAPs definition is a real-time task with a hard deadline, the latter being the beginning of the subsequent frame. If grant scheduling, grant arrangement and grant allocation can actually be implemented by independent sub-tasks interworking concurrently according to a pipeline scheme, the time that can be dedicated by each sub-task

to accomplish its work in every frame is basically doubled. This is because the grant scheduling module can start working on subsequent frames with respect to the grant arrangement and the grant allocation ones which are still defining the current one. Therefore, either less powerful processing hardware can be used inside the BS, with the same scheduling algorithm, or with the same hardware, more complex scheduling algorithms can be implemented.

3.1 Grant Scheduler

Within the PIPER framework, the grant scheduling sub-task can be viewed as an independent functional module whose goal is to guarantee QoS to admitted connections.

Figure 5 shows the grant scheduler architecture for QoS support. Each downlink connection has a packet queue (or *queue*, for short) at the BS (represented with solid lines). In accordance with the set of QoS parameters and the status of the queues, the *BS downlink scheduler* selects from the downlink queues, on a frame basis, the next SDUs to be transmitted to SSs. On the other hand, uplink connection queues (represented in Fig. 5 with dashed lines) are virtual and resemble those residing at SSs. As discussed in Chapter 2 bandwidth requests are used on the BS for estimating the residual backlog of uplink connections. In fact, based on the amount of bandwidth requested (and granted) so far, the *BS uplink scheduler* estimates the residual backlog at each uplink connection and allocates future uplink grants according to the respective set of QoS parameters and the (virtual) status of the queues. However, as already introduced, although bandwidth requests are per connection, the BS nevertheless grants uplink capacity to each SS as a whole. Thus, when an SS receives an uplink grant, it cannot deduce from the grant which of its connections was intended for by the BS. Consequently, an SS scheduler must also be implemented within each SS MAC in order to redistribute the granted capacity to the SSs connections. However, the implementation of the SS scheduler is out of the scope of this work and, thus, we will not discuss this point further.

Based on the structure reported in Fig. 5, one can envisage several different solutions for implementing the downlink and uplink scheduling functions. In fact, many scheduling algorithms have been put forward in the literature to sup-

port QoS in wired and wireless networks. For example, the Earliest Deadline First (EDF) (LL73) can be adopted to guarantee a bounded latency. Wongthavarawat *et al.* (WG00) proposed a solution based on EDF to serve rtPS connections. Specifically, each rtPS connection plays the role of a task in a real-time operating system, whose deadline is assumed to be equal to its maximum latency. This solution has been shown to provide rtPS connections with bounded delays, provided that the traffic is filtered through a token bucket shaper with rate equal to the Minimum Reserved Traffic Rate. More recently, additional simulation studies with the same architecture have been presented (CJW05; WG03), which confirm the original results obtained. Alternative approaches can be found within the class of latency-rate scheduling algorithms (SV98) as they are particularly suited for providing QoS guarantees in terms of rates. However, in general, any of the above mentioned scheduling strategies can not be applied straightforwardly as downlink and uplink schedulers in IEEE 802.16. In the rest of this section we thus describe how to adapt scheduling strategies present in the literature to cover the grant scheduling function.

Specifically, without loss of generality, within the class of latency-rate scheduling strategies, we consider the Deficit Round Robin (DRR) (SV96) to be a good candidate as the downlink scheduler to be implemented at the BS. The latter combines the ability of providing fair queueing, in the presence of variable length packets, with the simplicity of implementation.

With DRR, each connection i has a *quantum* (ϕ_i) that is the average number of bytes that will be scheduled during each round while it is backlogged.[1] Additionally, connection i stores a *deficit counter* that is used to keep track of the number of bytes that could not be scheduled because they did not fit into the remaining amount of service. For instance, assume that $\phi_i = 100\ B$ and there are three PDUs of size 60 B waiting for transmission. When the round of connection i comes, it will only be able to transmit one of those PDUs, and the remaining 40 B are accounted as *deficit* that can be exploited in future rounds. In this case, connection i will then be able to send two PDUs in the next round. As a result of this mechanism, the maximum amount of outstanding service of any connection is upper bounded by its maximum packet size. Note that if fragmentation

[1] We define a connection as *busy* (or *backlogged*) when it has one or more buffered SDUs awaiting transmission. Connections that are not busy are said to be *idle*.

is enabled for that connection, it is always possible to transmit the head-of-line PDU (or parts thereof) provided that there is room for the MAC header, plus fragmentation sub-header and CRC (if present). In order to serve connections in a proportional manner with respect to their minimum reserved rates, the following equality should hold for any two connections i, j with minimum reserved rates R_i, R_j, respectively:

$$\frac{\varphi_i}{\varphi_j} = \frac{R_i}{R_j}. \tag{3.1}$$

Therefore the *quantum* of connection i can be computed as

$$\varphi_i = \frac{R_i}{\sum_{k \in \mathcal{C}} R_k} F_{DRR}, \tag{3.2}$$

where $\mathcal{C}$ is the set of admitted connections, and F_{DRR} is the target DRR round duration, in bytes. The latter is a system parameter that can be tuned to adjust the scheduler responsiveness vs. the computational complexity. In fact, higher *quanta* allow the BS to produce the schedule of the forthcoming frame with less DRR iterations. However, a busy connection might be forced to wait a longer interval for its turn to come, which increases delays. Finally, since DRR is a work-conserving scheduling algorithm, while not all connections are backlogged at the same time the busy connections consume the remaining capacity. However, they do so in a fair manner by sharing it proportionally to their *quanta*, i.e., minimum reserved rates.

DRR assumes that the size of the head-of-line PDU is known at each connection queue, thus it cannot be used by the BS to schedule transmissions in the uplink direction. In fact, with regard to the uplink direction, the BS is only able to estimate the overall amount of backlog of each connection, but not the size of each backlogged packet. Thus one may think to select the Weighted Round Robin (WRR) (KSC91) instead, which belongs to the class of rate-latency scheduling algorithms as well. Basically, WRR works the same way as DRR, but connections do not keep a deficit counter. Therefore, any capacity not used by a connection in a round will not be compensated in future rounds, which can lead to unfairness among connections with different sizes of PDU. However, this problem becomes negligible if SSs exploit fragmentation to maximize the utilization of their uplink grants, which is very likely to be the case in IEEE 802.16. Even though

there are sophisticated schedulers in the literature that enjoy more advanced theoretical properties than WRR, e.g., the Weighted Fair Queueing (WFQ) (PG93), which is a well-known approximation of the General Processor Sharing (GPS) ideal scheduler, the improvement in terms of fairness and scheduling latency becomes negligible in IEEE 802.16 due to frame-based transmissions. This has been shown in a simulation study which compared WFQ and WRR (GOS04).

It is worth noting that service differentiation between different scheduling services requires appropriate setting of the minimum reserved rates, i.e., DRR's *quanta*. Since this problem is not the primary focus of this work we will not discuss it in more detail. The interested reader can found additional details in (CELM06; CELM07).

So far, our discussion has been rather general. In fact, we actually did not mention any reference to the specification of the IEEE 802.16 standard. Our line of reasoning has been totally abstract from the technological details of IEEE 802.16, e.g., we adopted well-known scheduling algorithms present in the literature for wired networks. This is one of the main advantages of the PIPER framework. However, besides the high flexibility in the selection of the specific grant scheduling strategy, a particular attention must be paid to defining the grant arrangement and allocation sub-task according to the constraint imposed by the standard. For instance, when the system operates in FDD, the BS must always schedule HD-SSs so that their transmissions and receptions do not overlap in time. In closed-loop system design, one may think to run the grant scheduling process, obtain the resulting schedule, and then check, *a posteriori*, whether the half-duplex constraint is respected or not. This information is then fed back to grant scheduler which must take the appropriate actions. However, with this approach, not only the system incurs a higher computational overhead, but also the grant scheduling output might be greatly affected by the half-duplex constraint. The PIPER approach is, instead, to design solutions in an open-loop pipelined manner by considering the system constraints *a priori* in the grant arrangement and grant allocation subtasks. In Chapter 4, we will describe in details how the HD-SS problem can be solved by the grant allocation sub-task by means of an algorithm, namely *HDA*, provided that the grant arrangement sub-task respects necessary, though simple, conditions.

Chapter 4

PIPER in IEEE 802.16d with OFDM

Design is the management of constraints.
— Dino Dini (1965 -)

This chapter describes how to address the resource allocation problem in IEEE 802.16d with OFDM according to the PIPER framework. We propose and evaluate algorithms for the grant arrangement and grant allocation sub-tasks so as to provide, with grant scheduling described in Sec. 3.1, a complete resource allocation solution for the IEEE 802.16d standard to implement. Without loss of generality, we consider IEEE 802.16d to operate in FDD with HD-SSs since the half-duplex constraint makes the system configuration more complex, hence, grant arrangement and grant allocation more challenging.

Figure 6 shows an example of a possible allocation for a generic frame with index n in IEEE 802.16d obtained through the PIPER framework pipelined processing. As described in Sec. 3.1, the grant scheduler subtask output consists of a list of connection grants to be advertised by the BS. The grant arrangement sub-task is then responsible for grouping the connection grants into SSs grants by also taking into account the half-duplex constraint. Finally, grant allocation will define the frame layout, i.e., the MAPs content, exploiting the Half-Duplex Allocation (HDA) algorithm which will be described in Sec. 4.2. As can be seen, the final layout of the uplink and downlink sub-frame satisfies the half-duplex

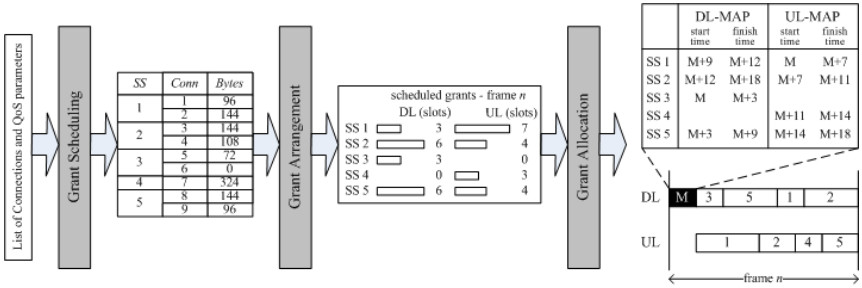

Figure 6: PIPER in IEEE 802.16d with OFDM and HD-SS.

constraint since SSs are never scheduled for transmitting and receiving simultaneously. In the following, we will show how we can always guarantee this condition without breaking the PIPER open-loop design structure.

4.1 Grant Arrangement

The main task of grant arrangement in IEEE 802.16d is to group the connection grants addressed to an SS into a single SS' grant which will be then advertised. This simple operation is actually fundamental to reduce the overhead due to the MAPs. In fact, should any connection of an SS represent a grant to be advertised, an IE for each SSs connection should be added to MAPs. Additionally, as discussed in Sec. 2.2.2, a physical preamble must be pre-pended to each advertised grant. Therefore, the greater the number of advertised grants, the lower the available frame capacity for data transmission. Then, grouping the connection grants belonging to the same SS appears to be necessary rather than recommended to keep a reasonably low system overhead. The standard does not provide any other specification which can be exploited so that the system overhead can be further reduced.[1] However, additional constraints are imposed by the physical layer specification and by the frame structure. First, since the grant scheduler is unaware of the underlying frame structure and capacity, in general, the overall scheduled slots can exceed the current frame capacity. Therefore, the

[1]Note that, the IEEE 802.16e standard amendment with OFDMA provides instead several mechanisms which can be exploited to reduce the system overhead as will be discussed in Sec. 5.1.

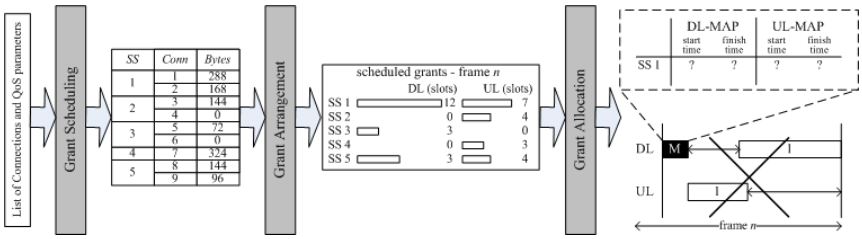

Figure 7: Unfeasible grant allocation due to improper grant arrangement.

grant scheduling sub-task must provide some buffering capabilities to keep track of the grants which can not be served within the current frame. Furthermore, it may happen that the duration of the grants, as determined by the grant scheduler, is such that there is no way of allocating them in the current frame without violating the half-duplex constraint. As an example, if the grant scheduler devised to grant in a frame both an uplink and a downlink grant to the same HD-SS, and the sum of the duration of the two grants exceeds the frame duration, the grant allocator is clearly not able to produce any valid MAP. Such an example is illustrated in Fig. 7, where the uplink and downlink grants addressed to SS 1 would overlap in time regardless of the allocation algorithm. It is worth noting, as we prove numerically in Sec. 4.3, that the case of scheduling a station both in uplink and downlink in the same frame, is a typical case when QoS guarantees must be provided. The grant arrangement sub-task is thus responsible, in each frame, for producing a set of grants which makes the grant allocation always feasible.

Many solutions can be devised to work around this problem. On the one hand, a solution could be to abandon the pipeline approach, and implement the scheduling function with algorithms that jointly define the duration and start time of each grant. By looking at Fig. 6, this latter approach could be described as to produce the final maps through a number of successive iterations involving the grant arrangement and allocation subtasks, based on the exchange of intermediate results between each other. However, this approach would also mean to leave out the advantages of the PIPER approach. On the other hand, one could try to find *sufficient* conditions to impose to the grant arrangement sub-task so that the grant allocator is always able to find a feasible grant allocation. For example, one could trivially impose that an HD-SS can only be granted bandwidth in one

direction in each frame. This would *avoid* the problem of satisfying the half-duplex constraint.

However, such sufficient conditions should be carefully selected, since they may greatly affect the capability of the grant scheduler to provide the admitted connections with the required QoS guarantees. We could formulate such a requirement by stating that, within the PIPER framework, the sufficient conditions to impose to the output of the grant arrangement process should be as loose as possible. On the other hand, it is straightforward to identify *necessary* conditions for the grants' duration defined by the grant scheduler, so that grant allocation could be feasible. Specifically, such *necessary* conditions are that the overall bandwidth, both downlink and uplink, granted to each HD-SS in each frame must not exceed the frame duration.[2]

In the next section we provide a formal proof that *the above mentioned necessary conditions are also sufficient* to guarantee that scheduled grants can be allocated in the respective sub-frames, without violating the half-duplex constraint. In this respect, such sufficient conditions are the loosest ones, because also necessary, i.e., they are essential to any other set of conditions that could be devised. Furthermore, based on this proof, we propose an algorithm for the grant allocator, namely the Half-Duplex Allocation algorithm (HDA), which is always able to arrange the scheduled grants in the respective sub-frames, provided that sufficient conditions are met by the grant arrangement sub-task. Therefore, HDA can be employed in the grant allocator whatever the algorithm implemented in the grant scheduler is, in complete consistency with the pipeline approach of the PIPER framework.

4.2 Grant Allocation

In this section, we first investigate the grant allocation sub-task in IEEE 802.16 in FDD with HD-SS and then we propose a solution for this sub-task. To this aim, we formally devise sufficient conditions to be met by grant arrangement, which guarantee that time allocation of grants will always be feasible without violating the half-duplex constraint, and we propose a grant allocation algorithm, namely,

[2]Without loss of generality, we refer to the case in which the uplink allocation start time is equal to the frame duration, and thus uplink and downlink sub-frames perfectly overlap in time.

the Half-Duplex Allocation algorithm (HDA)[3], which eventually allocates in the frame the grants scheduled to both FD-SSs and HD-SSs as output by the grant arrangements sub-task. Furthermore, we prove HDA to be: i) optimal, in the sense that, if there is a feasible allocation of a set of scheduled grants, then HDA is always able to find it; and ii) computationally efficient, since grant allocation is completed in $\mathcal{O}(n)$ steps, where n is the number of grants.

First, in Sec. 4.2.1 we introduce the notation and assumptions that will be used in the rest of the section. Then, in Sec. 4.2.2, we derive the *necessary* conditions that the grant arrangement sub-task has to enforce so that the grant allocation is feasible. Then we prove that these *necessary* conditions are also *sufficient* for a set of unicast[4] grants addressed to HD-SSs to be allocated in a frame without violating the half-duplex constraint. The proof of one of the main theoretical results, namely Lemma 1, allows us to define the HDA algorithm, which is described next. Section 4.2.3 illustrates the actual HDA procedure. To simplify the notations, we assume in the following that all grants are addressed to HD-SSs — hence, we refer to them as SSs for short — and the uplink allocation start time is equal to the frame duration. The solutions for the general case, in which FD-SSs are also present, and uplink and downlink sub-frames are not aligned in time, are reported in Sec. 4.2.4 and Sec. 4.2.5, respectively.

4.2.1 Definitions and Assumptions

We define the *available frame duration* (T, in time units) as the frame duration minus the duration of MAC control messages. The latter are broadcasted by the BS at the beginning of the downlink sub-frame for management purposes, and do not convey MAC SDUs. MAC control messages include the UL- and DL-MAP, as well as several messages as described in Sec. 2.1. In any case, all SSs need to listen to those messages, whose transmission durations may vary frame by frame. Thus, the overlapping portion of the uplink frame cannot be used for uplink grants, which is accounted for by the above definition of T.

[3]HDA is under a patent owned by Nokia Corporation (United States Application 20060245380).

[4]The results reported in this section cannot be straightforwardly extended to the case of multicast connections. However, the main challenge with multicast is scheduling grants taking into account the different physical profiles of the SSs which participate to a multicast session, since they dynamically adapt their modulation and coding according to physical layer measurements. Hence, we consider unicast connections only.

Symbol	Definition
n	number of SSs scheduler in the current frame
T	frame duration
u_i	durationg of the uplink grant to SS i
d_i	durationg of the downlink grant to SS i
s_i^u	start time of the uplink grant to SS i
f_i^u	finish time of the uplink grant to SS i
s_i^d	start time of the downlink grant to SS i
f_i^d	finish time of the downlink grant to SS i

Table 3: HDA glossary.

Without loss of generality we assume that any SS is scheduled exactly one grant for each direction per frame. In fact, should an SS be scheduled more than one grant in the same direction, all of them could be equivalently aggregated into a single grant whose duration is the sum of the originally scheduled grants' durations. This operation is performed by the grant arrangement sub-task in the PIPER framework. On the other hand, should no grant be scheduled to an SS in one direction, we equivalently assume that a grant of null duration is scheduled instead. We also assume that the grant duration includes any protocol overhead, such as the synchronization preamble.

Let n be the number of SSs scheduled in the current frame. The grant allocated to SS i, within the frame of available duration T, is uniquely identified by any two out of the following three quantities: the start time s_i^x, the finish time f_i^x, and the duration x_i. where $x \in \{d, u\}$ represents either the downlink or the uplink direction. It is straightforward to derive the relationships among the three quantities as follows:

$$\begin{cases} f_i^x &= |s_i^x + x_i|_T \\ s_i^x &= |f_i^x - x_i|_T \\ x_i &= |f_i^x - s_i^x|_T \end{cases},$$

where the modulo operator $|x|_T$ is defined based on the Euclid's Theorem as reported in (Bou92), i.e., $|x|_T = x - \lfloor x/T \rfloor \cdot T$. The notation is summarized in Table 3 and illustrated in Fig. 8.

We now provide a set of definitions and preliminary results, which formalize the problem of grant allocation.

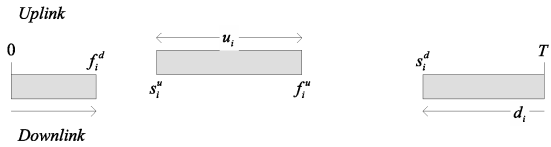

Figure 8: HDA notations.

Definition 1. A set $U = \{(s_i^u, u_i)\}$ ($D = \{(s_i^d, d_i)\}$) of uplink (downlink) allocated grants is said to be feasible iff, for any time instant $t \in [0, T]$, there exists at most one grant $(s_j^u, u_j) \subseteq U$ ($(s_j^d, d_j) \subseteq D$), $1 \leq j \leq n$, such that

$$\left|t - s_j^u\right|_T < u_j \qquad (\left|t - s_j^d\right|_T < d_j). \tag{4.1}$$

Inequality (4.1) means "the time instant t lies between the start and the end times of the uplink (downlink) grant addressed to SS j." Therefore, the definition above states in a formal way the following intuitive concept: an allocation is feasible iff the uplink and downlink allocated grants of any SS do not overlap in time.

Proposition 1. A set U of uplink grants is feasible iff $\left|s_i^u - f_j^u\right|_T + u_i + u_j \leq T$, $1 \leq i, j \leq n$, $i \neq j$. A set D of downlink grants is feasible iff $\left|s_i^d - f_j^d\right|_T + d_i + d_j \leq T$, $1 \leq i, j \leq n$, $i \neq j$.

Proof. The proof is straightforward considering that:

1. $\left|s_i^u - f_j^u\right|_T$ is the time interval between the end of one grant and the beginning of the next one;

2. u_i is the grant scheduled starting from s_i^u onward, u_j is the grant scheduled starting from f_j^u backward;

3. if the sum of the three is not greater than T then the grants u_i and u_j do not overlap.

The same line of reasoning also applies to downlink grants ■.

From Proposition 1, if U (D) is a feasible set of uplink (downlink) grants, then $\sum_i u_i \leq T$ ($\sum_i d_i \leq T$).

Definition 2. A pair $\{U, D\}$ of uplink and downlink sets is said to be feasible iff, for any SS i and time instant $t \in [0, T]$ such that $|t - s_i^u|_T < u$, it is $\left|t - s_i^d\right|_T \geq d_i$. In other words, the uplink and downlink grants for the same SS do not overlap in time, hence they do not violate the half-duplex constraint.

The following results can be easily proved.

Proposition 2. A pair $\{U, D\}$ of uplink and downlink feasible sets is feasible iff

$$\left|s_i^d - f_i^u\right|_T + u_i + d_i \leq T, \qquad 1 \leq i \leq n. \tag{4.2}$$

Thus $\{U, D\}$ is not feasible if any of the following conditions is false:

$$\begin{cases} \sum_{i=1}^n u_i & \leq & T \\ \sum_{i=1}^n d_i & \leq & T \\ u_i + d_i & \leq & T \qquad 1 \leq i \leq n \end{cases}. \tag{4.3}$$

Proposition 3. Given a feasible pair $\{U, D\}$, any pair $\{U_t, D_t\}$ such that $U_t = \left\{\left(|s_i^u + t|_T, u_i\right)\right\}$ and $D_t = \left\{\left(\left|s_i^d + t\right|_T, d_i\right)\right\}$, $t \in \mathbb{R}$, is also feasible. Thus, a feasible pair $\{U, D\}$ is uniquely identified except for a constant value.

4.2.2 Theoretical Results

The following Lemma will be used to prove Theorem 1.

Lemma 1. Let U be a feasible set of uplink grants. Let

$$\tilde{D} = \left\{d_k \in \mathbb{R} \middle| d_k \geq 0, 1 \leq k \leq n\right\}$$

be a set of n non-negative real numbers, and let X be defined as follows:

$$X \triangleq \left\{ x \in \mathbb{R} \middle| \max_i \left(\sum_{k=1}^{i} d_k - s_i^u \right) \leq x \leq \min_i \left(\sum_{k=1}^{i-1} d_k - f_i^u + T \right) \right\}. \tag{4.4}$$

If $\sum_{k=1}^n d_k \leq T$, and $u_i + d_i \leq T$, $1 \leq i \leq n$, then $X \neq \emptyset$.

Proof. Without loss of generality we assume that $i < j \Leftrightarrow s_i^u < s_j^u$. Furthermore, based on Proposition 3, we assume that $s_1^u = 0$, which yields $s_i^u < f_i^u$ for any i.

Proving the thesis is equivalent to prove that

$$\min_i \left(\sum_{k=1}^{i-1} d_k - f_i^u + T \right) - \max_i \left(\sum_{k=1}^{i} d_k - s_i^u \right) \geq 0,$$

i.e., for any i, j, $1 \leq i, j \leq n$, it must be

$$\sum_{k=1}^{j-1} d_k - f_j^u + T - \left(\sum_{k=1}^{i} d_k - s_i^u \right) \geq 0.$$

We consider three possible cases.

Case 1: $i > j$. It is

$$\sum_{k=1}^{j-1} d_k - f_j^u + T - \left(\sum_{k=1}^{i} d_k - s_i^u \right) = T - \sum_{k=j}^{i} d_k + s_i^u - f_j^u \geq 0,$$

by hypothesis, since $\sum_{k=j}^{i} d_k \leq \sum_{k=1}^{n} d_k \leq T$ for any $i > j$, and $s_i^u \geq f_j^u$ because of the feasibility of U. □

Case 2: $i < j$. It is

$$\sum_{k=1}^{j-1} d_k - f_j^u + T - \left(\sum_{k=1}^{i} d_k - s_i^u \right) = T - \left(f_j^u - s_i^u \right) + \sum_{k=i+1}^{j-1} d_k \geq 0,$$

since $f_j^u > s_i^u$ for any $i < j$, and $f_j^u - s_i^u \leq T$ by assumption. □

Case 3: $i = j$. It is

$$\sum_{k=1}^{i-1} d_k - f_i^u + T - \left(\sum_{k=1}^{i} d_k - s_i^u \right) = T - u_i - d_i \geq 0,$$

by hypothesis, for any i, which concludes the proof. ■

Finally we prove that the necessary conditions that must be met by any set of grants addressed to SSs in order to be feasible are also sufficient.

Theorem 1. Let U be a feasible set of uplink grants, and let $\{d_i\}$ be a set of grants. A set of downlink grants D, such that the pair $\{U, D\}$ is feasible, always exists iff $\sum_{j=1}^{n} d_i \leq T$, and $u_i + d_i \leq T$, $1 \leq i \leq n$.

Proof.

If part (sufficient condition). Without loss of generality we assume that $i < j \Leftrightarrow s_i^u < s_j^u$. Furthermore, based on Proposition 3, we assume that $s_1^u = 0$, which yields $s_i^u < f_i^u$ for any i. Let us consider a set of downlink grants D, such that

$$s_i^d = \left| \sum_{k=1}^{i-1} d_k - x \right|_T \tag{4.5}$$

for any $x \in X$, where X is defined according to (4.4). Note that the hypothesis of Lemma 1 are satisfied, hence $X \neq \emptyset$, i.e., the downlink set of grants is well-defined. It is straightforward to prove that D is feasible.

To complete the proof we show that the pair $\{U, D\}$ is feasible using Proposition 2. By substituting (4.5) into (4.2) we obtain:

$$\left|s_i^d - f_i^u\right|_T + u_i + d_i = \left| \sum_{k=1}^{i-1} d_k - f_i^u - x \right|_T + u_i + d_i. \tag{4.6}$$

Now, by definition of X, it follows that:

$$\sum_{k=1}^{i} d_k - s_i^u \leq x \leq \sum_{k=1}^{i-1} d_k - s_i^u - u_i + T,$$

for any $x \in X$. Hence x can be rewritten as

$$x = \sum_{k=1}^{i} d_k - s_i^u + y, \tag{4.7}$$

provided that $0 \leq y \leq T - u_i - d_i$. By substituting (4.7) into (4.6) we obtain:

$$\begin{aligned} &\left|s_i^d - f_i^u\right|_T + u_i + d_i = \\ &= \left| \left(\textstyle\sum_{k=1}^{i-1} d_k - \sum_{k=1}^{i} d_k \right) + (s_i^u - s_i^u) - u_i - y \right|_T + u_i + d_i \\ &= \left|-d_i - u_i - y\right|_T + u_i + d_i \\ &= T - d_i - u_i - y + u_i + d_i \\ &= T - y \leq T \end{aligned} \quad . \tag{4.8}$$

Since (4.8) holds for any i, $1 \leq i \leq n$, it completes the proof. □

Only if part (necessary condition). It immediately follows from Proposition 1 and Proposition 2. ■

4.2.3 HDA Algorithm

Based on the constructive proof of Lemma 1, we devised an algorithm, namely HDA, to allocate a set of uplink and downlink grants $\{u_i\} \bigcup \{d_i\}$ such that the half-duplex constraint is satisfied, provided that the necessary conditions are met. HDA, whose correctness is proved by Theorem 1, consists of the following three steps:

Step 1. Set the start and finish time of the uplink grant of each SS i as follows:

$$\begin{cases} s_1^u &= 0 \\ f_1^u &= s_1^u + u_1 \end{cases} \quad \begin{cases} s_i^u &= f_{i-1}^u \\ f_i^u &= s_i^u + u_i \end{cases} .$$

In other words, place the uplink grants contiguously from the beginning of the uplink sub-frame. This initial allocation of uplink grants will not be changed in the next steps.

Step 2. Temporarily set the start and finish time of the downlink grant of SS i as follows:

$$\begin{cases} s_1^d &= 0 \\ f_1^d &= s_1^d + d_1 \end{cases} \quad \begin{cases} s_i^d &= f_{i-1}^d \\ f_i^d &= s_i^d + d_i \end{cases} .$$

Step 3. The allocation so far may not comply with the half-duplex constraint. Select an offset $\bar{x}$ that will be used to update the allocation of the downlink grants, so that the resulting allocation complies with the half-duplex constraint. Left-shift each downlink grant in a circular way (i.e., modulo T) by $\bar{x}$. To do so, update the start and finish times of the downlink grant of each SS i as follows:

$$\begin{cases} s_1^d &= |-\bar{x}|_T \\ f_1^d &= s_1^d + d_1 \end{cases} \quad \begin{cases} s_i^d &= |f_{i-1}^d|_T \\ f_i^d &= |s_i^d + d_i|_T \end{cases} .$$

Based on Lemma 1 we can select $\bar{x}$ in the range X defined by (4.4), which is proved to be non-empty. It is worth noting that *any* value in X provides the same guarantees, in terms of feasibility of a set of input grants, as any other value in that range. For instance, in pseudo-code below, we chose $\bar{x}$ as the lower bound of X, i.e., $\bar{x} = \min\{X\} = \max_i \left(\sum_{k=1}^{i} d_k - s_i^u \right)$.

```
   /* initialization */

 1 sum = 0
 2 x = 0

   /* step 1 */

 3 for ( i = 1 ; i <= n ; i++ ) {
 4     if ( i == 1 ) s_u[1] = 0     /* start time  */
 5     else          s_u[i] = f_u[i - 1]
 6     f_u[i] = s_u[i] + u[i]       /* finish time */
 7 }

   /* step 2 */

 8 for ( i = 1 ; i <= n ; i++ ) {
 9     if ( i == 1 ) s_d[1] = 0     /* start time  */
10     else          s_d[i] = f_d[i - 1]
11     f_d[i] = s_d[i] + d[i]       /* finish time */
12 }

   /* step 3: selection of x */

13 for ( i = 1 ; i <= n ; i++ ) {
14     sum += d[i] - s_u[i] + s_u[i - 1]
15     x = max ( x, sum )
16 }

   /* left-shift downlink grants */

17 for ( i = 1 ; i <= n ; i++ ) {
18     if ( i == 1 ) s_d[1] = ( s_d[1] - x ) mod T
19     else          s_d[i] = ( f_d[i - 1] ) mod T
20     f_d[i] = ( s_d[i] + d[i] ) mod T
21 }
```

Figure 9: Pseudo-code of HDA.

The pseudo-code of HDA is reported in Fig. 9. After the initialization of local variables `sum` and `x` (1–2), the final allocation of the uplink grants is settled by placing them contiguously at the beginning of the uplink sub-frame (3–7). This is done by setting the start time of the first uplink grant to zero (4), and that of any other grant to the finish time of its previous grant (5). Finish times, on the other hand, are computed as the sum of the start time and the grant duration (6). The same operation is carried out for downlink grants (8–12). Unlike the uplink one, downlink allocation is only temporary. At step 3, we compute the value of the offset `x` that will be used to left-shift downlink grants to produce an allocation which complies with the half-duplex constraint (13–16). Finally, the downlink temporary allocation is "left-shifted" in a circular way, by means of the modulo (`mod`) operator (17–21), as defined above (Bou92). More specifically, the start time of the first downlink grant is brought forward of `x` time units (18), then the start time of any other downlink grant is set to the finish time of its previous grant (19). Finish times, on the other hand, are computed as the sum of the start time and the grant duration (20).

The computational complexity of HDA can be derived as follows. The body of each `for` loop includes elementary operations that are performed at constant time, with respect to the number of grants to allocate (say, n). Since each for `loop` consists of $\mathcal{O}(n)$ iterations, the computation complexity of the whole procedure is $\mathcal{O}(n)$.

4.2.4 Extension for Mixed HD-SSs and FD-SSs

When both HD- and FD-SSs are present in the network and require grant allocation in the same frame, a straightforward solution would be to treat FD-SSs as they were half-duplex, and therefore apply HDA to both of them. However, with such an approach, we would constrain the sum of the downlink and uplink grants addressed to each FD-SS to be smaller than the frame duration, which is not a necessary condition for them. A more efficient implementation of the grant allocator, which still complies with the PIPER framework, follows:

1. *(grant arrangement)* arrange uplink and downlink grants to HD- and FD-SSs, such that (i) the sum of the uplink and downlink grants of any HD-SS is smaller than or equal to the frame duration, and (ii) the sum of all uplink

(downlink) grants is smaller than or equal to the frame duration;

2. *(grant allocator)* allocate grants addressed to HD-SSs using HDA, as though they were the only ones;

3. *(grant allocator)* allocate grants addressed to FD-SSs in the remaining portion of uplink and downlink sub-frames.

In Step 1 the duration of downlink grants addressed to FD-SSs does not include the time to transmit a physical preamble, which is never required to resynchronize them. Step 2 always succeeds because the sufficient conditions expressed by (4.3) hold for the subset of scheduled grants addressed to HD-SSs. The remaining capacity in Step 3 will be certainly sufficient for FD-SS grants, since the overall sum of grant durations — including both HD-SSs and FD-SSs — in each direction, as produced by the grant arrangement, cannot exceed the frame duration. Finally, note that it is possible to allocate FD-SSs uplink grants at the exact beginning of the uplink sub-frame, i.e., partially overlapping with the MAC control messages broadcasted in downlink by the BS, thus making use of that bandwidth which is never available to HD-SSs.

4.2.5 Extension for Non-Time-Aligned Uplink and Downlink Sub-Frames

HDA can also be extended to take into account the general case in which the uplink and downlink sub-frames are not perfectly aligned in time, i.e., the uplink allocation start time is equal to $T' < T$. Figure 10 illustrates this scenario, where sub-frames are numbered according to the map relevance, i.e., the downlink sub-frame x hosts the UL-MAP containing the timetable of uplink grants in sub-frame x.

Let us consider uplink and downlink sub-frames n. Let t_0 be the start time of downlink sub-frame n, which is thus *logically* split into two sections: the first section begins at t_0 and ends at $t_0 + T'$, when the uplink sub-frame n begins; the second section begins at $t_0 + T'$ and ends at $t_0 + T$. Therefore, the duration of the first section is T', and that of the second section is $T - T'$. Note that the uplink and downlink grants of the first section belong to frame n and $n-1$, respectively, whereas those of the second section belong to frame n only. For this reason, it is

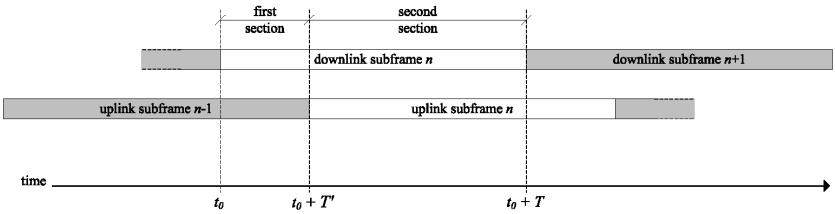

Figure 10: HDA with non-time-aligned uplink and downlink sub-frames.

not possible to run a single instance of HDA over the whole downlink or uplink sub-frame. Therefore, an instance of HDA is carried out for each section, where the necessary and sufficient conditions for a feasible grant allocation in (4.3) are modified as follows:

$$\begin{cases} \sum_{i=1}^{n_1} u_i^1 & \leq & T' \\ \sum_{i=1}^{n_1} d_i^1 & \leq & T' \\ u_i^1 + d_i^1 & \leq & T' \quad 1 \leq i \leq n_1 \end{cases} \qquad \begin{cases} \sum_{i=1}^{n_2} u_i^2 & \leq & T - T' \\ \sum_{i=1}^{n_2} d_i^2 & \leq & T - T' \\ u_i^2 + d_i^2 & \leq & T - T' \quad 1 \leq i \leq n_2 \end{cases},$$

where n_1 and n_2 are the number of grants in the first and second section, respectively. In our pipeline framework, an instance of the grant scheduler and the grant arrangement sub-task are carried out for each section as well. Finally, note that the duration of MAPs is subtracted from the available frame duration in the first section only.

4.3 Performance Evaluation

In this section we show the effectiveness of our solutions for IEEE 802.16d as developed within the PIPER framework. We will evaluate the performance of both FD-SSs and HD-SSs under realistic traffic conditions through simulation. Results are obtained using a C++ implementation of the solutions proposed in this work. Specifically, we i) adopted DRR/WRR(SV96; KSC91) for downlink/uplink grant scheduler (see Sec. 3.1), ii) implemented the grant arrangement sub-task such that the necessary and sufficient conditions described in Sec. 4.2 are always respected and, iii) employed the HDA algorithm as grant allocator. The evaluated scenarios

Parameter	Value(s)
Channel bandwidth	7 MHz
OFDM symbol duration	34 μs
Cyclic prefix duration	2 μs
Frame duration	5 ms, 10 ms, 20 ms
Uplink allocation start time	one frame duration
Minimum contention window	16 slots
Maximum contention window	1024 slots
Modulation	QPSK, 16-QAM, 64-QAM
FEC type	RS-CC

Table 4: Simulation and network configuration parameters.

are based on the business opportunity of employing IEEE 802.16 as the last-mile Internet access technology for residential and SME subscribers, as envisaged by the WiMAX forum (For04a). The presented results have been partially published in (BCE+07).

First we explain the simulation environment in detail. The simulations were carried out by means of a prototypical simulator of the IEEE 802.16 protocol. The simulator is event-driven and was developed using C++. The MAC layer of the BS and SSs are implemented, including all procedures and functions for uplink/downlink data transmission, and uplink bandwidth request/grant.

We assume ideal channel conditions, i.e., no packet corruption is due to the wireless channel impairment. This allows us to get insight into the the ability of the PIPER framework to exploit the mechanisms that are provided by the IEEE 802.16 MAC to manage data and multimedia traffic, regardless of any specific assumptions on the channel characteristics. Finally, we analyze the system while in a steady-state, when the set of admitted connections does not change over time. Thus, we do not assess the performance of the signaling protocol between the BS and SSs for establishing new connections and the admission control procedures at the BS.

The system parameters used in the simulation analysis are reported in Table 4. The physical layer parameters are those envisaged by the WiMAX forum in (For04b), and currently employed by manufacturers producing IEEE 802.16-compliant devices (e.g., (com)). As already pointed out, the IEEE 802.16 takes on adaptive modulation and coding to adjust data transmission to different chan-

#SSs	64QAM-3/4	16QAM-3/4	BPSK-1/2	QPSK-3/4
10	1972.8	1479.6	328.8	986.4
20	1828.8	1371.6	304.8	914.4
30	1684.8	1263.6	280.8	842.4
40	1540.8	1155.6	256.8	770.4
50	1396.8	1047.6	232.8	698.4
60	1252.8	939.6	208.8	626.4
70	1108.8	831.6	184.8	554.4
80	964.8	723.6	160.8	482.4
90	820.8	615.6	136.8	410.4

Table 5: Raw bandwidth, in kB/s, available in uplink with a 5 ms frame depending on the burst profile employed and the number of SSs that served per frame.

nel conditions, which may depend on a number of factors, including path loss, shadowing and multi-path fading, and interference from nearby SSs. Therefore, in our simulation scenarios we consider a mix of SSs employing different modulation schemes. Specifically, based on the results presented in (Hoy05), which was derived by assuming that SSs are uniformly distributed in a circular cell, with the BS placed in the center, the number of SSs employing QPSK modulation is assumed to be twice as much as the number of SSs employing 16-QAM modulation, which in turn is twice as much as the number of SSs employing 64-QAM modulation. The amount of raw bandwidth, i.e., not including any MAC overhead, available in uplink is reported in Table 5 as a reference for the throughput measures below.

The simulation analysis was carried out using the method of independent replications (LK00). Specifically, the simulation of each scenario was repeated 20 times, by initializing the random number generators with different seeds. The duration of each run was 1200 s, with a warmup period of 360 s, during which measures were not collected. In all the simulation runs, we estimated the 95% confidence interval of each performance measure. The confidence interval is not drawn whenever it is negligible.

4.3.1 Traffic Models and Workload Characterization

Different types of traffic sources are used in the simulation scenarios. The data traffic is modeled as a Web source. We used the Web source model, namely Web

	Web exponential	
	Inter-arrival	Packet size
Distribution	Exponential	Pareto with cut-off
Parameters	$\lambda = 5$ s	$\alpha = 1.1$ $k = 4.5$ kB $m = 2$ MB
Average rate	25 kb/s	

	VoIP		
	Inter-arrival	Packet size	ON/OFF period
Distribution	Deterministic	Deterministic	Exponential
Parameters	20 ms	66 B	$\lambda_{ON} = 1.34$ s $\lambda_{OFF} = 1.67$ s
Average (peak) rate	11.7 kb/s (26.4 kb/s)		

Table 6: Workload characterization. Web and VoIP sources.

exponential, reported in (Mot00). Multimedia traffic is evaluated by means of Voice over IP (VoIP). Specifically, we model VoIP traffic as an ON/OFF source with Voice Activity Detection (VAD). Packets of 66 bytes are generated only during the ON periods, at fixed intervals of 20 ms, so that a GSM Adaptive Multi Rate encoder at 3.3 kB/s is simulated (Pre01). The duration of the ON and OFF periods is distributed exponentially (Bra69). Table **??** shows the characterization of the two traffic models. Web traffic is served by means of the BE scheduling service whereas VoIP traffic employ the rtPS one.

The performance was assessed using the following metrics. The *transfer delay* (or delay, for short) is defined as the time interval between the instant when a packet arrives at the MAC layer of the source node (SS/BS) and the time that this packet is completely delivered to the next protocol layer of the destination node (BS/SS). Both the average and the Cumulative Distribution Function (CDF) are estimated. The *delay variation* is the difference between the 99th percentile of the delay and the packet transmission time, i.e., the time it takes for a packet of minimum length to be transmitted over the air. This metric is of paramount importance for VoIP traffic, and should be kept as small as possible so as to satisfy the quality of service perceived by the users of VoIP applications. Web traffic, on the other hand, though interactive, has less stringent delay requirements and is thus evaluated by means of the average delay. *Buffer occupancy* is the

number of bytes awaiting transmission at a connection buffer, either downlink or uplink. Lastly, we measured the number of SSs served in both directions within the same frame, i.e., the number of SSs that have at least one downlink and one uplink grant in the downlink and uplink sub-frames occurring at the same time. The latter helps to quantify the impact of the half-duplex constraint on the grant scheduler operation. We set up each replication of the simulation scenario as follows. We provided each SS with a random number of connections. The number of connections was selected according to a geometric distribution with ratio 0.5 truncated at 9. In terms of the workload, we assumed that each connection carries aggregate traffic from a random number of basic sources, either Web or VoIP depending on the scenario, whose characterization and average rate are reported in Table **??**. The number of sources was sampled from a geometric distribution with ratio 0.5 truncated at 9. Furthermore, the MCS of each SS was uniformly distributed in the set QPSK-3/4, 16-QAM-1/2, 64-QAM-2/3, which entails encoding 24, 48, and 72 bytes per OFDM symbol, respectively. With regard to SA, each SS was randomly placed into the odd or even set, with equal probability.

4.3.2 Simulation Results

Since, to the best of our knowledge, there is no previous work which explicitly deals with the allocation of grants for HD-SSs within IEEE 802.16 frames, we compare the performance of HDA to that obtained with FD-SSs (FD). The analysis below consists of varying the following network parameters and workload configuration: (i) type of traffic (i.e., VoIP or Web); (ii) number of SSs from 10 to 60; (iii) frame duration (i.e., 5 ms, 10 ms, or 20 ms).

We start with the case of VoIP traffic, with frame duration equal to 20 ms. Table 7 reports the average number of SSs that are served in both directions within the same frame. Recall that, in general, the output of the BS grant schedulers (i.e., DRR/WRR) depends on several factors, e.g., traffic characterization, QoS parameters. Additionally, the grant arrangement sub-task is subject to the half-duplex constraint, which only holds in the HD-SS case. Therefore, the difference between the results obtained in the FD case, when grant arrangement is not subject to the half-duplex constrain, and HDA, when the necessary and sufficient conditions of Sec. 4.2.2 are enforced by the grant arrangement sub-task, can be

#SSs	HDA	FD
10	5.137	5.137
20	10.37	10.38
30	15.40	15.53
40	20.74	20.76
50	26.03	26.03
60	31.10	31.20

Table 7: Average number of SSs served in both directions within the same frame.

considered as a measure of the impact of the half-duplex constraint on the operation of the grant scheduler. As can be seen, when HDA is employed as the grant allocator, the half-duplex constraint does not affect significantly the grant scheduler operation, i.e., the difference between HDA and FD results is negligible irrespective of the number of SSs in the network.

We now analyze the delay variation of uplink and downlink connections, which is reported in Fig. 11. As can be seen, each curve consists of two regions. In the first region, i.e., when the number of SSs is smaller than or equal to x (in uplink $x = 38$, in downlink $x = 42$), the delay variation is almost stable. In the second region, i.e., when the number of SSs increases further, there is a steep increase of the delay variation, which significantly degrades the quality perceived by the users of VoIP applications. This is clearly due to the capacity of uplink and downlink sub-frames being saturated by the overall offered load, i.e., the system is in overloaded conditions. Since any HD-SS can be served in any frame by using HDA, as showed in Table 7, the HDA and FD curves are almost overlapping.

The delay variation of downlink connections with HDA is slightly greater than that with FD. This is because HD-SSs experiment a slightly worse service than FD-SSs as described in Table 7. Nevertheless, the half-duplex constraint does not impact significantly on the BSs grant scheduler since the performance degradation of HDA with respect to FD is negligible. As far as the uplink is concerned, the HDA and FD curves almost overlap. The small difference in Fig. 11 is because FD-SSs are able to exploit the full duration of the uplink subframe, including the interval while the BS is transmitting the DL- and UL-MAP, which is instead unavailable for HD-SSs.

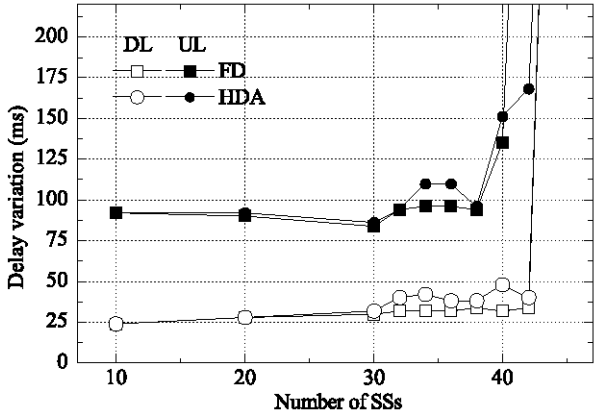

Figure 11: Delay variation of VoIP connections vs. number of SSs.

In Fig. 12 and Fig. 13 we report the CDF of the delay of downlink and uplink connections, respectively, with 30 SSs, i.e., before the steep increase of the delay variation. The same qualitative behavior is experimented with any number of SSs in either directions, whose results are thus not reported. As can be seen, in both downlink and uplink, the HDA and FD curves are almost overlapping, the latter incurring a slightly smaller delay than the former due to the half-duplex constraint, as discussed above. Moreover, the bandwidth request mechanism justifies the minimum delay of about 30 ms experienced in both cases.

To conclude the analysis of VoIP traffic, we now investigate how the frame duration impacts on the delay variation of downlink and uplink connections with HDA, which is plotted in Fig. 14. As can be seen, the shorter the frame duration, the smaller the delay variation. In downlink, this is because any SDU enqueued at the BS after the DL-MAP has been sent has to wait the next frame before it can scheduled for transmission. In uplink, the frame duration has an even stronger impact than in downlink because of the bandwidth request mechanism. In fact, the delay of an SDU is the sum of the latency that the SS experiences to obtain an opportunity to send a bandwidth request to the BS, which becomes greater as the frame duration increases, plus the time that is needed by the BS to schedule an uplink grant to the SS, which is at least one frame duration.

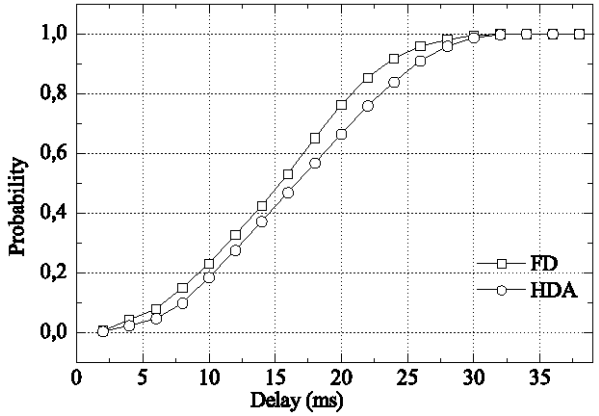

Figure 12: CDF of the delay of downlink VoIP connections, with 30 SSs.

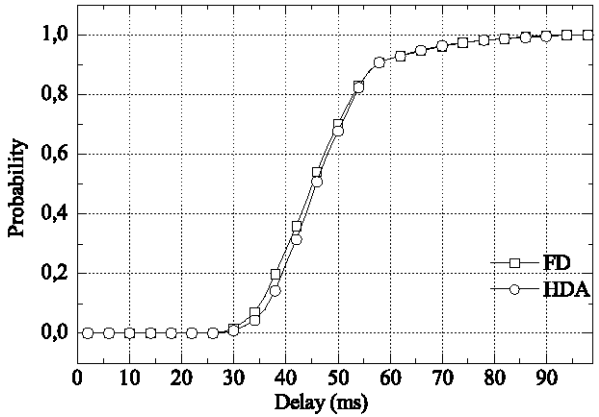

Figure 13: CDF of the delay of uplink VoIP connections, with 30 SSs.

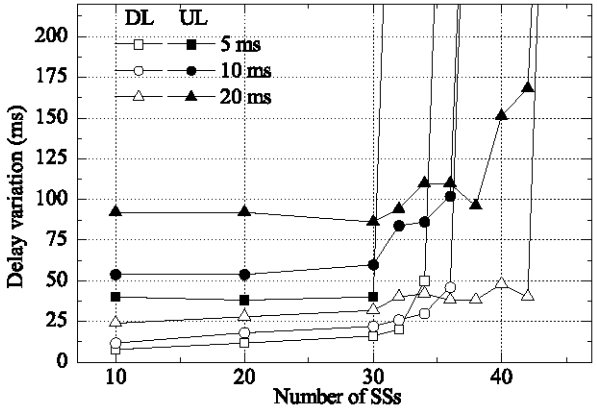

Figure 14: Delay variation of VoIP connections vs. number of SSs. HDA only.

However, in both directions, the sub-frame capacity saturates with a smaller number of SSs when the frame duration becomes shorter, which can be explained as follows. In downlink this is because there is a fixed control overhead per frame, due to the transmission of the DL-MAP and UL-MAP messages and the long physical preamble. This overhead consumes an increasing amount of net capacity available for data transmission when the frame duration becomes shorter. With regard to uplink, with shorter frame durations the BS is more timely to react to the bandwidth requests sent by the connections, which decreases the delays but increases the number of uplink grants that each connection is scheduled, on average, in a time unit. This increases the overhead due to physical preambles.

We now discuss the results obtained with Web traffic, with frame duration equal to 20 ms. Figure 15 shows the average delay of downlink and uplink connections. Basically the same conclusions of VoIP traffic can be drawn in this case, i.e., the difference between the FD and HDA curves is negligible and slightly increases when the offered load increases. Furthermore, unlike the VoIP case, the uplink sub-frame saturates with a smaller number of SSs than the downlink sub-frame. This is because part of the uplink sub-frame is reserved by the BS to the transmission of bandwidth requests in a contention-based manner, which reduces the capacity available in the uplink with respect to downlink.

We conclude the analysis of Web traffic by analyzing the CDF (Fig. 16) of the

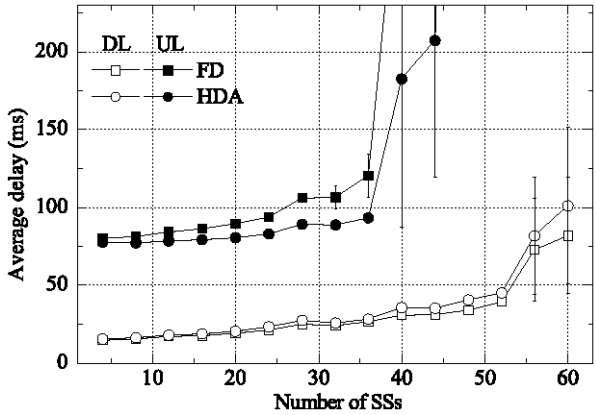

Figure 15: Average delay of Web connections vs. number of SSs.

buffer occupancy of downlink and uplink connections before the steep increase of the average delay, i.e., with 36 and 52 SSs in the uplink and downlink direction, respectively. As can be seen, the difference between the HDA and FD curves is negligible under this metric too. Thus, HD-SSs served by a BS employing HDA are expected to have approximately the same requirements, in terms of their buffer size.

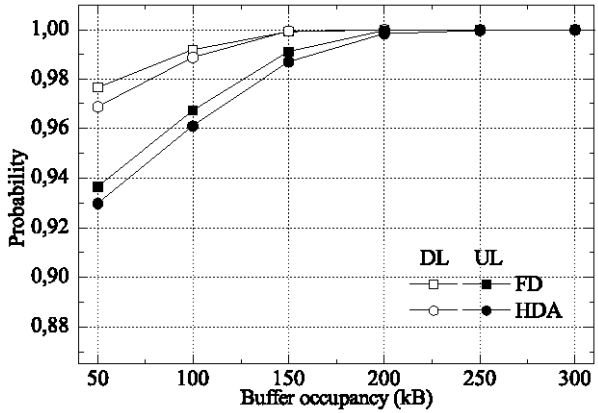

Figure 16: CDF of the buffer occupancy of Web connections, with 36 (52) SSs in uplink (downlink).

Chapter 5

PIPER in IEEE 802.16e with OFDMA

For every problem, there is one solution which is simple, neat and wrong.
— Henry Louis Mencken (1880-1956)

As already pointed out in Sec. 2.2.2, the inherent properties of the OFDMA air interface makes the resource allocation process even more challenging than with OFDM. In fact, the bi-dimensional frame structure poses additional constraints with respect to OFDM. A schematic description of the PIPER approach in IEEE 802.16e operating in TDD is reported in Fig. 17. Still, the grant scheduling sub-task complies with the specifications defined in Sec. 3.1. On the other hand, the grant arrangement sub-task distributes the selected grants into different groups according to specific arrangement strategies discussed in Sec. 5.1. Such groups are tentatively targeted to being mapped one-to-one to as many data regions to be allocated in the frame. These data regions are then fed to the last sub-task, i.e., grant allocation, which is responsible for defining the final map content, possibly re-arranging and/or re-sizing data regions to fit them into the OFDMA frequency and time domain. To this end, we propose an algorithm, namely Sample Data Region Allocation (SDRA), which performs grant allocation in a heuristic manner and is described in Sec. 5.2.1. SDRA also works when H-ARQ is enabled. We extensively evaluate our solution by means of Monte Carlo analysis in Sec. 5.3.

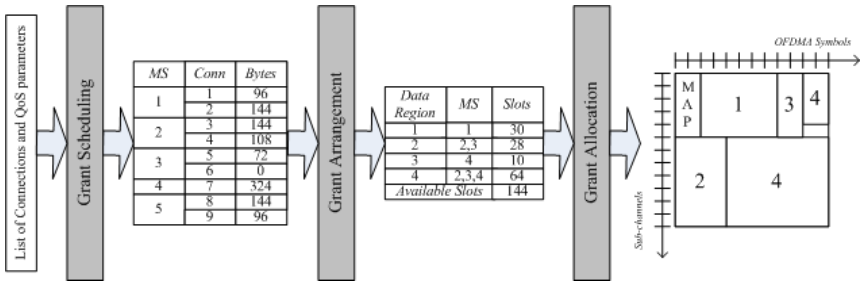

Figure 17: PIPER in IEEE 802.16e with OFDMA.

So far we have not distinguished between uplink and downlink in our discussion. However, in the uplink sub-frame, the IEEE 802.16e standard explicitly defines the procedures for the grant arrangement and the grant allocation sub-tasks whereas grant scheduling is left up to manufacturers. In fact, as already pointed out in Sec. 2.2.2, uplink data regions are explicitly defined by the standard as a number of contiguous slots, starting from the upper-left corner of the uplink sub-frame, allocated in row-wise order. Conversely, in the downlink sub-frame, all the three sub-tasks are left unspecified. Therefore, in the following, we will mainly refer to the downlink sub-frame allocation procedure unless otherwise specified. We will describe the standard procedures defined for the uplink direction whenever necessary so as to provide a term of comparison for the downlink sub-frame allocation.

5.1 Grant Arrangement

As mentioned above, different grant arrangement strategies, both in downlink and uplink sub-frames, may impact differently on the MAC control overhead. For the sake of clarity, we investigate how the different data region arrangement strategies affect the MAC control overhead in the downlink and the uplink sub-frames separately. First we focus on the downlink direction. Since the control messages signaling is in-band, reducing the DL-MAP size is beneficial as it allows more data to be conveyed into the same downlink sub-frame, thus possibly increasing the overall throughput while preserving QoS guarantees. IEEE 802.16e

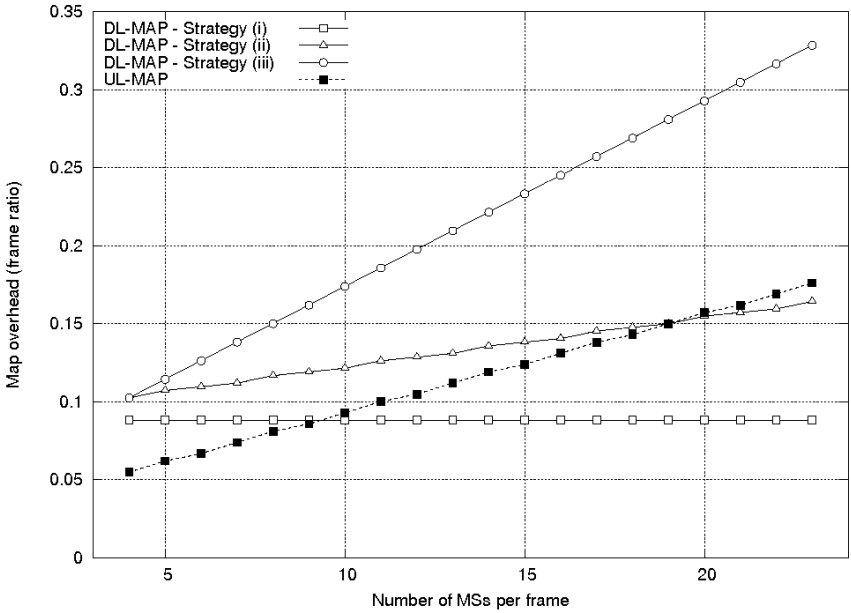

Figure 18: DL and UL map overhead with different scheduling strategies.

provides standard mechanisms for the purpose of reducing the control messages overhead. In fact, the DL-MAP size could be reduced by packing data addressed to MSs which employ a common MCS into a single data region. As mentioned in Sec. 2.2.2, for each IE, the BS is allowed to omit the list of MSs to which the PDUs are addressed, provided that the latter are all transmitted with the same MCS.

This may result in a considerable reduction of the DL-MAP size. However, this mechanism has the drawback that each MS is forced to decode all data regions transmitted with its own MCS[1], due to the lack of information with regard to the actual receivers of the data contained into the data region. Therefore, on average, an MS incurs greater energy consumption than needed. This might be a problem especially with mobile terminals, due to their limited energy capabili-

[1]Note that this does not violate the security functions of the IEEE 802.16 to ensure privacy of data transmitted over-the-air, since the payload of MAC PDUs can be encrypted so that only the intended recipient is able to retrieve the original data sent by the BS.

ties. On the other hand, energy wastage can be mitigated by explicitly specifying the lists of data region recipients in the IEs, which reduces the chance that an MS unnecessarily decodes the data region, at the cost of increasing the overall map overhead. In fact, in this case, an MS only decodes those data regions actually containing data addressed to it. However, an MS can still incur some energy wastage, due to the fact that a data region may also convey data addressed to other MSs with the same MCS. Yet, having exactly one MS for each data region completely avoids waste of energy, though at the cost of the highest overhead.

To evaluate the relevance of such overhead, we define the DL-MAP size theoretical lower bound (TLB), with respect to a set of data regions, as the size of the minimum DL-MAP needed to allocate them. Specifically, TLB is computed assuming that each data region never needs to be split. However, this assumption is in general optimistic, since there might not be any feasible allocation of all data regions without splitting one or more of them. The latter operation is performed by the grant allocation sub-task. In Fig. 18 we show the DL-MAP overhead, in terms of the ratio between TLB and the downlink sub-frame duration, versus the total number of MSs served in the downlink and uplink sub-frames. We assume that each MS is provided with one connection. Numerical results have been obtained with the following system parameters: the frame duration is 10 ms with an FFT size of 1024 and a channel bandwidth of 10 MHz; the downlink and uplink sub-frames consist of 28 and 18 OFDMA symbols, respectively; the number of sub-channels is 30; and MSs employ four different combinations of MCS. The curves reported in Fig. 17 are related to three different strategies for the definition of the DL-MAP: (i) MSs are grouped, in downlink sub-frame, per MCS without specifying the list of the downlink data region recipients; (ii) same as (i), but the list of the downlink data region recipients is specified;[2] (iii) each MS, in the downlink sub-frame, is assigned to only one downlink data region. As shown in Fig. 18, with strategy (i) the overhead remains constant with the number of MSs. In fact, the size of the DL-MAP does not depend on the number of MSs served in the downlink sub-frame, but it only depends on the number of different downlink MCSs currently used, i.e., four in this scenario. On the other hand, strategy (iii) exhibits an increasing overhead with the number of MSs per

[2]Note that since we assume that each MS has only one connection, the number of the data region recipients is equal to the number of MSs per frame.

frame which severely impacts on the downlink sub-frame capacity by consuming 34% (55% including the UL-MAP) of the sub-frame with 23 MSs. Intermediate results have been obtained with strategy (ii), where the list of the downlink data region recipients is also advertised within the IE relative to a specific MCS. Note that the UL-MAP overhead increases linearly with the number of MSs due to the presence of one IE for each MS. With regard to the uplink direction, the IEEE 802.16e standard explicitly requires the BS to add an IE to the UL-MAP for each MS served in the uplink sub-frame, regardless of the MCS of the MS. In other words, the UL-MAP size only depends on the number of MSs served in the uplink sub-frame. Therefore, unlike downlink, the BS cannot count on any mechanism of the standard to reduce the control messages overhead. Note that the UL-MAP overhead affects the capacity available for transmission in the downlink sub-frame, although it depends on the number of MSs served in the uplink sub-frame. For the sake of completeness, we also reported the UL-MAP overhead in Fig. 18. We stress the fact that grant allocation procedure of the uplink sub-frame is explicitly defined by the standard. Therefore, the UL-MAP curve only depends on the number of MSs actually scheduled to transmit data, i.e., it only depends on grant scheduling, in the uplink sub-frame since no grant arrangements are permitted, except that explicitly specified by the standard.

We can conclude that, in an IEEE 802.16e system, the overhead due to MAPs can significantly reduce the capacity available for transmitting data. This problem can be mitigated by employing a data region arrangement strategy which limits the number of IEs to be advertised.

5.2 Grant Allocation

We now describe the downlink grant allocation sub-task, which, in OFDMA, consists of selecting both the size, i.e., width and height, and the position, i.e., the OFDMA symbol and sub-channel offsets, of each data region. These two problems are discussed separately. With regard to data region shaping, in general, there are several ways to shape the same number of scheduled slots into a data region. For example, assume that the BS has to transmit eight slots of data to an MS. If only one data region is used, there are four sizes available: 2×4, 4×2, 1×8 or 8×1. Still, it is possible for the BS to employ multiple data regions, e.g.,

two data regions of four slots each. Finally, the BS can inflate the data regions so that more than eight slots are used, e.g., a single 3×3 data region, in which case part of the inflated data regions remains empty. However, not all alternatives are equivalent, since the MAC overhead varies depending on the fragmentation of MAC PDUs over various data regions, and the padding of slots that are not completely used for user data.

As far as positioning of data regions is concerned, this problem can be viewed as a bi-dimensional bin packing problem, where a number of items (i.e., data regions) with a specified width and height have to be packed into a fixed-size bin (i.e., the downlink sub-frame). This problem is well known in the literature of operations research and has been proved to be NP-hard (MT90). Therefore it is not feasible to implement exact methods at the BS such as the HDA algorithm proposed in Sec. 4.2, since positioning of data regions is a hard real-time task, with a deadline comparable to the frame duration. Additionally, no efficient heuristics to solve general bin packing problems are known. We thus argue that simple, yet effective, heuristics should be envisaged, by taking into account the inherent properties of IEEE 802.16. For instance, a joint approach between shaping and placing data regions might be employed, so that the size of items to be allocated is modified to broaden the space of solutions. A heuristic algorithm has been recently proposed in (BSKD06) which aims at optimizing the spatial efficiency of transmissions. Furthermore, the use of H-ARQ introduces an additional constraint when shaping data regions because H-ARQ data regions cannot be split at arbitrary boundaries. For example, assume that the BS decides to split an H-ARQ data region consisting of two H-ARQ sub-bursts each consisting of four slots into two data regions. The only feasible configuration is $\{4,4\}$, while $\{5,3\}$, $\{6,2\}$, $\{7,1\}$ would not fit both (un-fragmented) H-ARQ sub-bursts.

5.2.1 A Sample Data Region Allocation Algorithm

In this section, we propose an algorithm, that we name Sample Data Region Allocation (SDRA) algorithm, as a solution to the grant allocation problem in OFDMA, i.e., shaping and positioning downlink data regions in PUSC/FUSC zones. According to the PIPER framework, SDRA works independently of the specific grant scheduling algorithm and the grant arrangement strategy adopted.

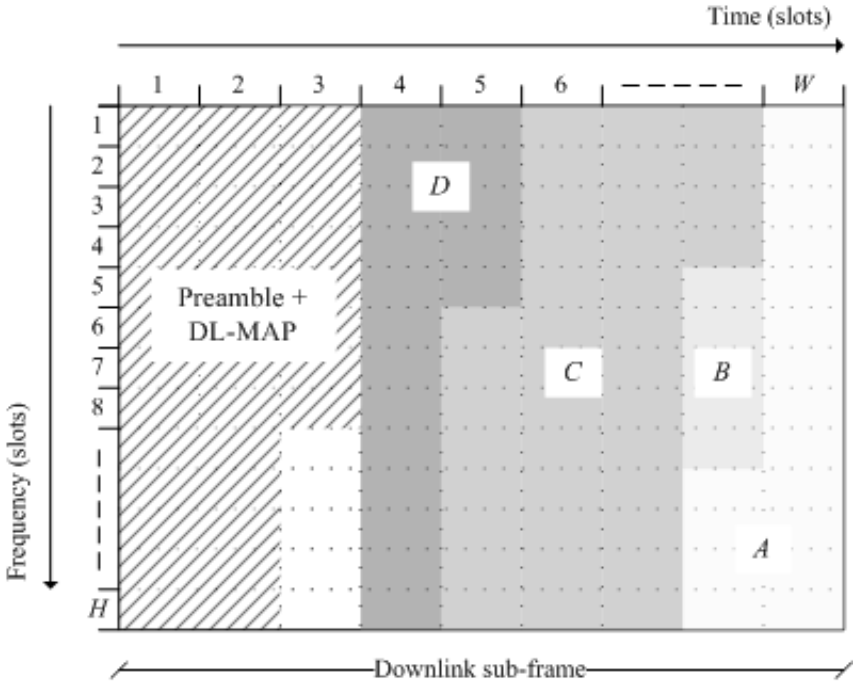

Figure 19: Example of non-H-ARQ allocation via SDRA.

Without loss of generality, we describe our algorithm in the case in which data addressed to MSs which employ a common MCS are grouped into a single data region (strategy (i) in Sec. 5.1). The SDRA algorithm is then extended to support H-ARQ as described at the end of this section. The performance of SDRA is assessed by means of Monte Carlo analysis in Sec. 5.3.

For the ease of readability, we describe the SDRA algorithm in the case of data transmitted in a FUSC zone only and exploit the notation of Fig. 19. The more general case of alternated PUSC/FUSC zones in the same downlink sub-frame can be trivially derived by taking into account that data regions in the PUSC zone can occupy the slots that lie below the DL-MAP, if any. In the following a slot is said to be allocated if the DL-MAP defines a data region which covers it. More formally, slot (i, j) is allocated if there is at least one IE such that

$x \leq i \leq x + w \wedge y \leq j \leq y + h$, where x, y are the time, sub-channel upper-left coordinates of the data region, respectively, and w, h are the data region dimensions in terms of its width and height, respectively. Note that an allocated slot might not contain any data actually. Moreover, we define the number of slots in a row of the downlink sub-frame as W; likewise, the number of slots in a column of the downlink sub-frame is H. Lastly, we assume without loss of generality that columns and rows are numbered starting from 1. SDRA is based on the following key concepts: (i) the order in which the data regions are passed to SDRA is preserved when they are allocated into the downlink sub-frame; (ii) allocation proceeds backwards in column-wise order, i.e., any slot of column i is not allocated until all the slots of columns $j, i + 1 < j \leq W$ are allocated. These design choices are motivated as follows. Because of (i) the scheduler is free to decide the priority of the downlink MAC PDUs that have to be transmitted, which can be exploited to assign to certain MAC PDUs a higher priority than others. Examples of MAC PDUs which may need a higher priority than others include: the UL-MAP and other MAC management messages; data that have not been allocated in previous frames; data belonging to admitted connections with strict QoS requirements, e.g., UGS connections. Additionally, a (partially) opportunistic approach could also be devised, where MAC PDUs of MSs enjoying higher transmission efficiency (i.e., less robust MCS) are more likely to be placed than the others. The impact of this choice on the network utilization is investigated in the performance analysis section. Furthermore, choice (ii) is motivated by the implementation concern that the DL-MAP grows in column-wise order starting from the beginning of the downlink sub-frame, according to the standard specifications. Therefore, by letting data filling up the sub-frame from the opposite direction, any grant needs only be allocated once and for all by the allocation procedure, as described below. We first describe the procedure to allocate a set of non-H-ARQ data regions, which are called pending data regions, while those actually placed into the downlink sub-frame are called allocated data regions. The procedure for H-ARQ data regions is described afterwards. Note that these procedures are described separately for the clarity of illustration, but they can be seamlessly integrated into a single allocation procedure to jointly allocate both non-H-ARQ and H-ARQ data regions. As already discussed in Sec. 2.2.2, we assume that any non-H-ARQ data region can be split into smaller data regions

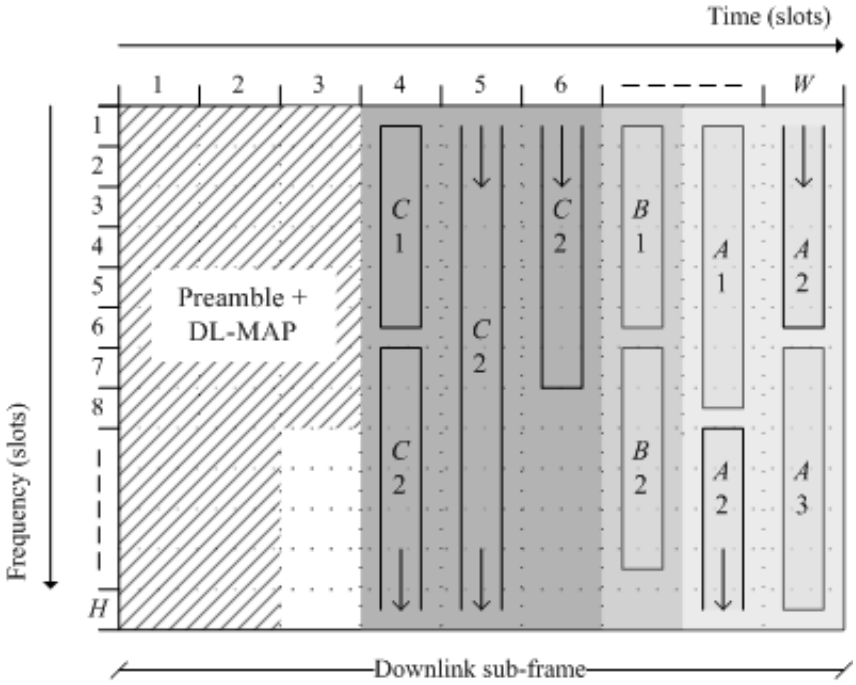

Figure 20: Example of H-ARQ allocation via SDRA.

arbitrarily, though at slot boundaries. Thus, since data regions are allocated in column-wise order, each data region can be split at most into three smaller data regions.[3] An example of the resulting allocation of four data regions, labeled A, B, C, and D, is depicted in Fig. 19.

As can be seen, the four input data regions produced eight data regions: A was split into two data regions, C into three, D into two, while B was allocated as a whole. Under the assumption of employing FUSC as the sub-carrier permutation, no more data can be allocated into the downlink sub-frame, because the free room that lies below the DL-MAP (column 3 in 19) cannot be exploited. The detailed allocation procedure is described by means of the pseudo-code in Ap-

[3]Therefore, the schedules should overprovision the amount of slots requested for each data region so as to include the MAC overhead due to (up to) two fragmented MAC SDUs. However, the quantitative impact on the network utilization is negligible.

pendix A of this chapter. As described in Sec. 2.2.2, unlike non-H-ARQ, H-ARQ data regions cannot be arbitrarily split into pieces, even though fragmentation of MAC SDUs is enabled in the IEEE 802.16 network. The allocation procedure described above is thus modified accordingly so as to support H-ARQ data region allocation. Specifically, each data region is always allocated as a rectangle with height equal to the number of rows available. The number of columns is then set to the minimum value such that the sum of the sizes of all the H-ARQ sub-bursts contained into the data region is smaller than or equal to the data region size. This way, up to $H - 1$ slots can remained unused in each data region. A sample allocation of three H-ARQ data regions, A with three sub-bursts, and B and C with two sub-bursts, is depicted in Fig. 20.

There is a major difference between the non-H-ARQ and the H-ARQ versions of SDRA. With H-ARQ, each input data region produces at most one allocated data region, while non-H-ARQ data regions can be split into up to three data regions. However, the latter can always be allocated provided that there is free room in the downlink sub-frame left by the DL-MAP. On the other hand, a H-ARQ data region can contain a sub-burst that does not fit into the remaining space, and thus needs to remain unallocated, while some sub-bursts of other H-ARQ data regions may be successfully allocated. Therefore, the allocation procedure with H-ARQ must proceed even when one or more sub-bursts are discarded, which is never the case with non-H-ARQ. The detailed allocation procedure is described by means of the pseudo-code in Appendix A of this Chapter.

5.3 Performance Evaluation

The performance evaluation is carried out by means of Monte Carlo analysis. Specifically a set of input numerical instances is defined based on the system and network configuration parameters reported below. An input numerical instance is defined as a list of data regions, each with a specified size and MCS; with H-ARQ, the data region also contains the list of H-ARQ sub-bursts. The exact procedure to generate a numerical instance is described in the Appendix B of this Chapter.

The parameters are classified into static and dynamic ones: static parameters are set in a deterministic manner and are used to derive the dynamic parameters;

Parameter name	Symbol	Type	Possible values
Sub-carrier permutation		static	PUSC, FUSC
H-ARQ support	h	static	enabled, disabled
Downlink sub-frame size (slots)	s	static	17 30, 17 10 (PUSC) 34 16, 34 5 (FUSC)
Number of MCSs		static	seven, three
Data region order		static	random (RND), More Robust First (MRF), Less Robust First (LRF)
Target offered load	tol	static	[0, 1]
Target percentage of VoIP users	v	static	0.2, 0.4, 0.6, 0.8
Size of VoIP PDUs, bytes	S_v	static	72
Size of BE PDUs, bytes	S_b	static	192
MCS (number of bytes/slot)	mcs	dynamic	*Seven MCSs* QPSK-1/2 (6), QPSK-3/4 (9), 16QAM-1/2 (12), 16QAM-3/4 (18), 64QAM-1/2 (18), 64QAM-2/3 (24), 64QAM-3/4 (27) *Three MCSs* QPSK-1/2 (6), 16QAM-1/2 (12), 64QAM-1/2 (18)
User type	u	dynamic	VoIP, BE
Average number of BE PDUs per user	$1/p_{be}$	dynamic	1, 2, 4

Table 8: Numerical instance parameters.

dynamic parameters are sampled from a random distribution during the numerical instance generation, based on the set of static parameters. Each combination of static parameters is defined as a snapshot. The allocation procedure is then fed with several variations of the snapshot, by initializing the random number generator functions with different initial seeds. Performance metrics, defined in Sec. 5.3.1, are estimated for a given snapshot by averaging their respective values over all the numerical instances of that snapshot. No weighting is applied; therefore all numerical instances are assumed as equally probable states of the system, and confidence intervals can be derived using the standard method of independent replications (LK00). Confidence intervals are however not drawn whenever negligible with respect to the estimated average. The system and network configuration parameters are reported in Table 8.

As outlined in Sec. 2.2.2, a slot consists of a two-dimensional time/subchannel rectangle, whose exact dimensions depend on the sub-carrier permutation, i.e., PUSC or FUSC. This difference accounts for the varying size of the downlink sub-frames reported in Table 8. Additionally, the greater downlink subframe sizes, i.e., 17×30 and 34×16, correspond to the case of 10 MHz physical bandwidth, with a frame duration equal to 5 ms, and ratio between the downlink and uplink sub-frames set to $35/12$. The same profile is used for the cases 17×10 and 34×5, respectively for PUSC and FUSC, where a physical re-use of three is assumed. In any case, the downlink sub-frame size includes the slots used to transmit the DL-MAP, which is computed according to the standard specifications as a function of the number of data regions (both non-H-ARQ and H-ARQ) and H-ARQ sub-bursts (H-ARQ only). For the purpose of analysis, we define the *offered load* (in slots, ol_S) as the fraction of the overall bin size s required to be allocated as user data:

$$ol_S = \frac{\sum_{j=1}^{n} a_j}{s}$$

where a_j is the size of the *j-th* data region provided as input to the allocation procedure, and n is the number of data regions. On the other hand, the target offered load for a given numerical instance is the reference value used as a stopping criterion during the numerical instance generation (see below). It is worth noting that the offered load associated to a given instance may differ from the

target offered load, due to the integer nature of the numerical instance generation process. On the other hand, the offered load (in bytes, ol_B) is computed as the sum of the bytes conveyed by the data regions provided as input to the allocation procedure. As far as grant arrangement is concerned, we assume that all MAC PDUs directed to users with the same MCS are combined into the same data region. The number of MCSs employed depends on the static parameter specified in Table 1. Specifically, if seven MCSs are used, this means that each user is served with the MCS that best fits its current channel conditions, as reported by the MSs (EMP$^+$06). On the other hand, with three MCSs we simulate a system where the BS can transmit data to some MSs (i.e., the ones that would be served with the MCSs that are unavailable in the three MCSs case) using a more robust MCS. This approach leads to a slightly lower transmission efficiency for some MSs, but reduces the number of IEs that are advertised on average in the DL-MAP, and hence the map overhead.

Lastly, the order in which data regions are fed to the allocation algorithm is specified as a static parameter. Three possible approaches are defined:

- Random (RND): Data regions, each related to a different MCS, are shuffled in a random fashion.
- More Robust First (MRF): Data regions are sorted in decreasing robustness order. In other words, data that has the least transmission efficiency, in terms of the number of bytes conveyed per slot, has the highest chance of being allocated.
- Less Robust First (LRF): This is opposite approach than MRF, i.e., data regions are sorted in increasing robustness order.

The impact on the performance of both the ordering approach selected and the number of employed MCSs is evaluated in the performance analysis in Sec. 5.3.2.

5.3.1 Metrics

The following performance metrics are defined. The *carried load* (in slots, cl_S) is defined as the fraction of the bin size s that is allocated to user data as a result of the allocation algorithm. Note that slots which are formally allocated as a result of the grant allocation, but that are not actually used to transmit user data, are

not included in this computation. On the other hand, the carried load (in bytes, cl_B) is defined as the number of bytes that can be conveyed by the data regions resulting from the output of the allocation algorithm, depending on their MCS. Then, the *success probability* is defined as the probability that the carried load in slots (in bytes) is equal to the offered load in slots (in bytes). It measures the capability of the algorithm to find a solution which satisfies all the specified constraints. The *map overhead* is defined as the fraction of the bin size s that is allocated to transmit the DL-MAP message. The *unused slots per frame* is defined as the fraction of the bin size *s* that is not allocated for any transmission. The *padding overhead* is defined as the fraction of the bin size *s* that is allocated in any data region, as a result of map definition, but is not actually used for data transmission. This accounts for slots that are allocated so as to define data regions as rectangles, but are not needed to accommodate transmission foreseen in that data region. This metric is only meaningful when H-ARQ is enabled, as discussed in Sec. 5.2.1.

5.3.2 Results

In order to evaluate the impact of the system and network configuration parameters on the performance of the downlink MAC frame allocation alone, the results in this section have been obtained without taking into account the MAC overhead due to the management messages other than the DL-MAP, including the UL-MAP. Furthermore, in all the evaluated scenarios, we verified that the performance improvement with PUSC is negligible with respect to FUSC. I n other words, the gain due to the number of slots that lie below the DL-MAP, which cannot be allocated in FUSC mode, is quantitatively small in terms of the metrics defined in Sec. 5.3.1. Therefore, we only report the results with FUSC for clarity of illustration.

We start with the analysis of the downlink MAC frame allocation procedure without H-ARQ support, by evaluating the carried load (in slots, cl_S), the map overhead, and the success probability when the offered load (in slots, ol_S) increases from 0.2 to 1. For the ease of presentation, the carried load plotted also includes the map overhead. In Fig. 21 we first compare the results with different downlink sub-frame sizes, i.e., 34×16 and 34×5. As can be seen, a large

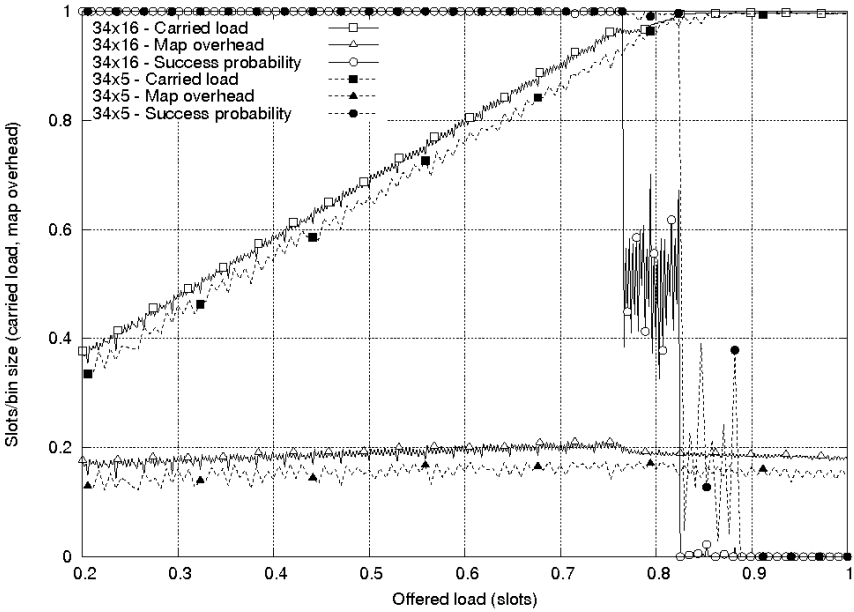

Figure 21: FUSC mode without H-ARQ. Carried load (in slots), map overhead, and success probability vs. the offered load (in slots), with different downlink sub-frame size, i.e., 34×16 and 34×5, seven MCSs and data regions randomly passed to the allocation algorithm.

portion of the downlink sub-frame (i.e., about 20% with a 34×16 sub-frame) is consumed by the transmission of the DL-MAP. Furthermore, this overhead does not significantly depend on the offered load. In fact, the size of the DL-MAP only depends on the number of IEs contained, i.e., data regions allocated, which varies in a narrow interval, i.e., between 1 and 7, which is the maximum number of MCSs. Note that the map overhead does not decrease linearly with the sub-frame size, because of the fixed information that needs to be conveyed in the DL-MAP, regardless of the number of data regions allocated. As far as the carried load is concerned, both the 34×16 and the 34×5 curves increase when the offered load increases until there is enough room in the downlink sub-frame to allocate the input data completely.

On the other hand, when the success probability becomes smaller than one,

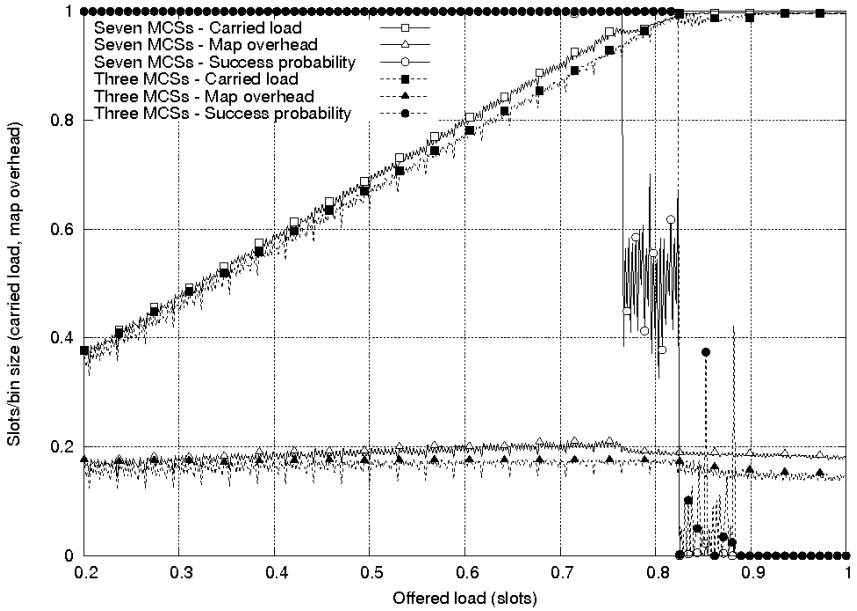

Figure 22: FUSC mode without H-ARQ. Carried load (in slots), map overhead, and success probability vs. the offered load (in slots), with different number of MCSs employed, i.e., seven and three, 34×16 downlink sub-frame size, and data regions randomly passed to the allocation algorithm.

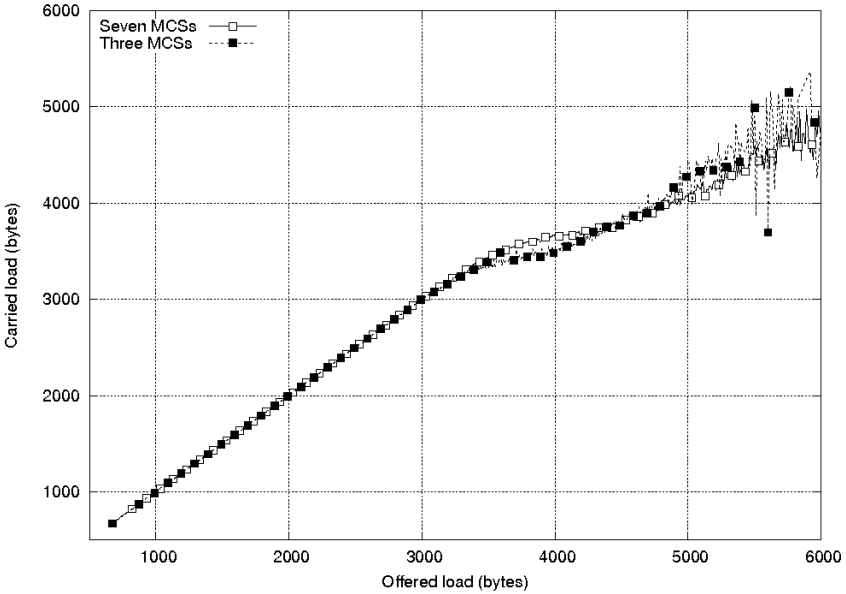

Figure 23: FUSC mode without H-ARQ. Carried load (in bytes) vs. offered load (in bytes), with different number of MCSs employed, i.e., seven and three, 34×16 downlink sub-frame size, and data regions randomly passed to the allocation algorithm.

the carried load saturates to a constant value. In other words, after some value of the offered load that depends on the size of the sub-frame, the latter is always completely allocated (i.e., the sum of the carried load and the map overhead is one), but some input bursts are still discarded (i.e., the offered load is greater than the carried load). Since this effect is due to the DL-MAP transmission overhead, the actual value of the offered load that saturates the sub-frame depends on the sub-frame size: the offered load with a 34×16 sub-frame saturates earlier than that with a 34×5 sub-frame. Note that there is a transient interval of the success probability where most, but not all, the numerical instances fail to be allocated. In the results above, all seven MCSs have been considered. This case is compared to that with three MCSs only in Fig. 22, with a 34×16 sub-frame. As can be seen, having a smaller number of MCSs reduces the map overhead, which in turn improves the success probability. In fact, with three MCSs the latter drops below one at a value of the offered load about 6% greater than that with seven MCSs.

However, the lower overhead due to MAPs does not improve significantly the performance, in terms of the carried load (in bytes) conveyed in the allocated regions, which is reported in Fig. 23 against the offered load (in bytes). This can be explained as follows.

With three MCSs only, some PDUs that can be transmitted with a less robust MCS (e.g., 16-QAM-2/3) are allocated instead by means of a less efficient MCS (e.g., 16-QAM-1/2). While this reduces the number of data regions allocated on average, and hence the map overhead, the average transmission efficiency is also lessened. Specifically, the plot in Fig. 23 exhibits three regions. First, when the offered load can be entirely allocated (i.e., when it is smaller than 3500 bytes), the carried load with both seven and three MCSs coincide with the former. Then, there is a region (i.e., with the offered load between 3500 bytes and 4500 bytes) where the carried load does not increase in a linear manner anymore. In this interval having seven MCSs yields a higher carried load, in terms of bytes, due to the higher transmission efficiency. Finally, when the offered load is greater than 4500 bytes, the effect due to the map overhead reduction with three MCSs overruns that of the transmission efficiency, which leads to improved performance with three MCSs only. Thus, while the number of slots allocated per sub-frame, reported in Fig 22 is always greater with a smaller number of MCSs, the performance in terms of bytes carried shows a trade-off depending on the input rate, in

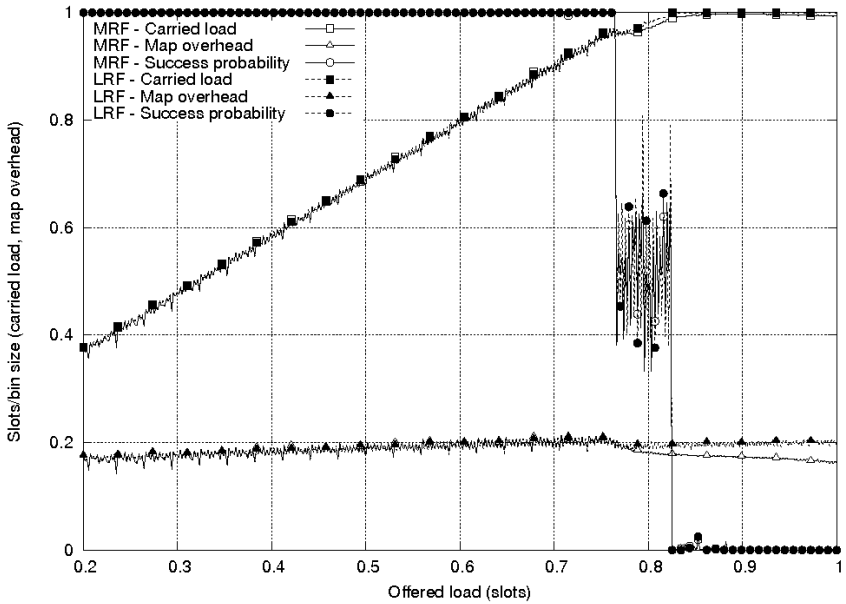

Figure 24: FUSC mode without H-ARQ. Carried load (in slots), map overhead, and success probability vs. the offered load (in slots), with different ordering of data regions depending on their MCSs, i.e., MRF and LRF, 34×16 downlink sub-frame size, and seven MCSs employed by the BS.

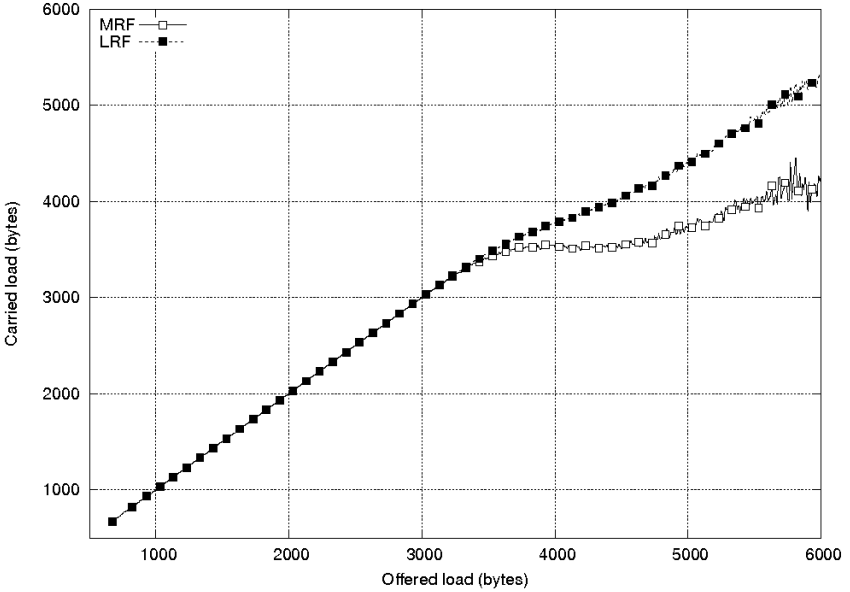

Figure 25: FUSC mode without H-ARQ. Carried load (in bytes) vs. offered load (in bytes), with different ordering of data regions depending on their MCSs, i.e., MRF and LRF, 34×16 downlink sub-frame size, and seven MCSs employed by the BS.

bytes, to the allocation algorithm.

We now investigate the performance in the cases where the data regions, before being fed to the allocation algorithm, are sorted according to the MRF and the LRF strategies described above. In this scenario, we consider seven MCSs. As can be seen in Fig 24, the specific ordering strategy does not affect significantly the performance in terms of carried load (in slots). In fact, with both the MRF and the LRF strategies all the curves (almost) overlap. However, beyond the value of about 0.8 of the offered load, the map overhead curves exhibit a divergent trend. Specifically, the MRF strategy entails an increasing smaller map overhead than the LRF one. This can be explained as follows. When the offered load is such that the success probability is smaller than one, i.e., the allocation algorithm fails to allocate all the data regions, the number of data regions allocated on average, and thus the map overhead, is greater with LRF than with MRF. In

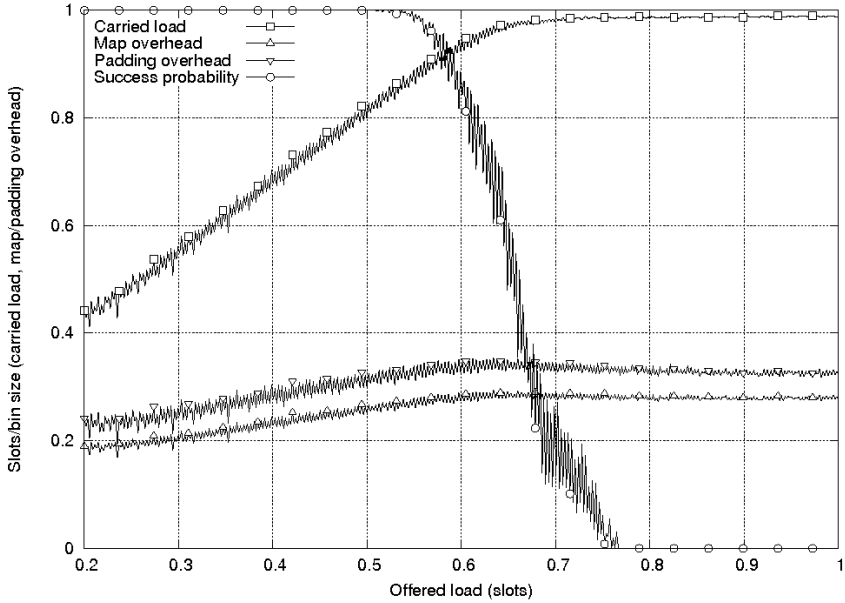

Figure 26: FUSC mode with H-ARQ. Carried load (in slots), map overhead, padding overhead, and success probability vs. the offered load (in slots), with 34×16 downlink sub-frame size, seven MCSs and data regions randomly passed to the allocation algorithm.

fact, the greater the transmission efficiency of an MCS, the smaller the size of the data region to convey the same amount of data. Therefore, allocating the data regions in decreasing robustness of MCS produces, on average, a smaller number of data regions which are actually allocated. This accounts for the overhead reduction in the MRF case when the value of the offered load is greater than 0.8.

Conversely, the higher the transmission efficiency, the higher the carried load, in terms of bytes, which can be achieved. This result is shown in Fig. 25 where the carried load (in bytes) versus the offered load is reported. With an offered load greater than 3500 bytes the MRF and LRF curves no longer overlap. The higher transmission efficiency obtained by the LRF strategy yields a carried load up to 5300 bytes while the MRF strategy reaches 4200 bytes at most. So far we have analyzed scenarios where the H-ARQ is not considered. We now extend our analysis to the case of FUSC sub-carrier permutation with H-ARQ.

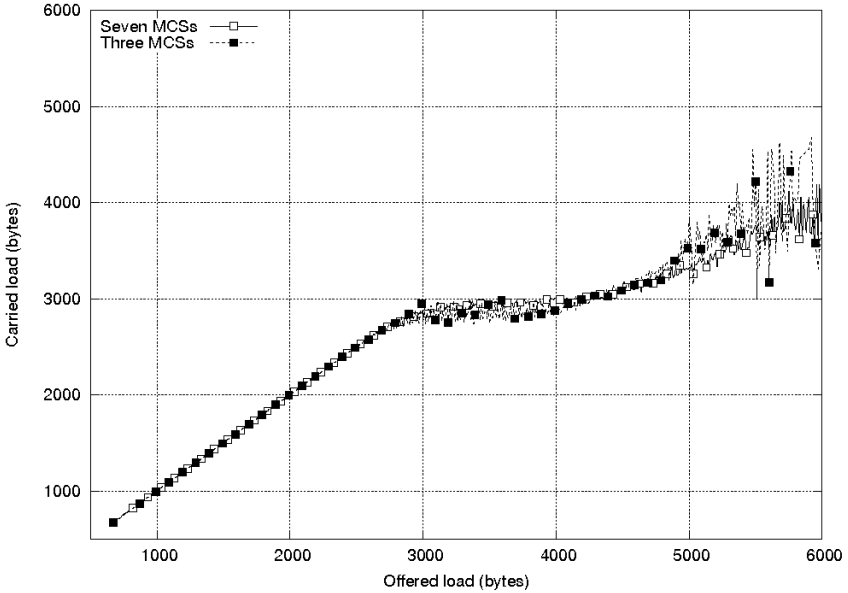

Figure 27: FUSC mode with H-ARQ. Carried load (in bytes) vs. offered load (in bytes), with different number of MCSs employed, i.e., seven and three, 34×16 downlink sub-frame size, and data regions randomly passed to the allocation algorithm.

Again, in Fig. 26, we report the carried load (in slots), the map overhead and the success probability versus the offered load (in slots) with 34×16 downlink sub-frames and seven MCSs. The padding overhead is also reporte as a measure of the capacity wasted due to the H-ARQ allocation algorithm. The same considerations of the results described earlier without H-ARQ still hold. However, remarkable is the impact of the padding overhead. The latter is comparable with the map overhead, reaching the value of about 0.3. As an effect of the padding overhead, note that the point in which the success probability starts dropping to zero is when the offered load reaches the value of about 0.5 which is actually much less than the value of about 0.75 obtained when H-ARQ is disabled (see Fig. 21).

The padding overhead also impacts on the carried load (in bytes), which is plotted in Fig. 27 against the offered load (in bytes). The results with both seven

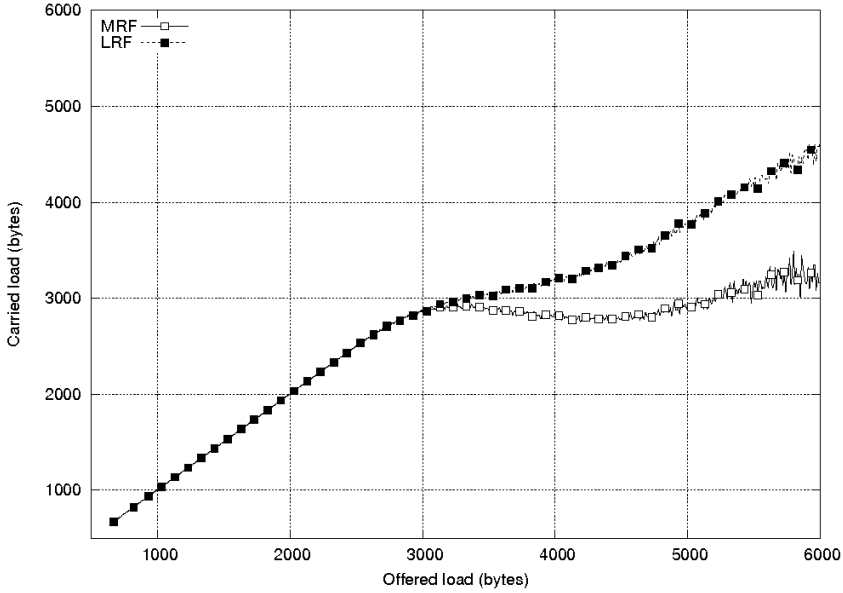

Figure 28: FUSC mode with H-ARQ. Carried load (in bytes) vs. offered load (in bytes), with different ordering of data regions depending on their MCSs, i.e., MRF and LRF, 34×16 downlink sub-frame size, and seven MCSs employed by the BS

MCSs and three MCSs are reported. While the curves exhibit a similar trend to the case with H-ARQ disabled (see Fig. 23), the quantitative results are much different. Specifically, the maximum value of carried load obtained with H-ARQ is 4000 bytes whereas it is 5000 bytes when the H-ARQ is disabled. Furthermore, as the offered load increases beyond the value of 3000 bytes the combined effect of the transmission efficiency and the map overhead described in Fig. 23 is smoothed by the presence of the padding overhead. Therefore, it is no longer possible to precisely identify the regions in which the curve with seven MCSs lies above the one with three MCSs and vice-versa. We can thus conclude that H-ARQ yields poorer performance, in terms of the carried load, due to the padding overhead.

Finally, in Fig. 28 we compare the carried load (in bytes) with the MRF and LRF strategies versus the offered load (in bytes). The results obtained are similar to those without H-ARQ, which have been reported in Fig. 25. In particular, note that the LRF curve lies above the MRF curve as soon as the offered load becomes greater than 3000 bytes.

Appendix A: SDRA Algorithm Pseudo-codes

In this section we report and describe the pseudo-codes of the SDRA algorithm both in the non-H-ARQ and the H-ARQ cases which have been informally introduced in Sec. 5.2.1.

Fig 29 reports the pseudo-code of SDRA without H-ARQ support. Data structures are initialized by creating an empty map (1) and a list of allocated data regions (2), and by setting the working variables w and h to the dimensions of the downlink sub-frame, respectively width and height (3-4). These variables will keep at each step of the procedure the coordinates of the next slot to be allocated. The allocation procedure is performed by placing one of the data regions in `list_pending` at each iteration of the main loop (7). The iteration terminates when either there are no more data regions to be allocated (7) or adding a new IE to the DL-MAP would make the latter overlap with already allocated data regions (13, 24, 33), whichever comes first. The size of the DL-MAP is updated via the `add_non_harq_region()` function. Newly allocated data regions are added to the `list_allocated` data structures, which consists of a

list of data region coordinates, expressed in terms of upper-left corner (i.e., starting column and row, respectively, in slots) and dimensions (i.e., width and height, respectively, in slots).

After the data region currently under allocation is extracted from the pending list (9), the allocation consists of (up to) three steps:

1. The first step (11-19) allocates a data region upwards starting from h into the remaining portion of the current column, i.e., w. The number of slots allocated is thus equal to h, provided that there are enough slots in the data region, otherwise the whole data region is allocated (e.g., B in Fig. 19). This step is skipped if the next slot to be allocated is at the bottom of downlink sub-frame, i.e., h is equal to H.

2. The second step (22-29) allocates a data region that spans over multiple columns (cols) (e.g., the second data region resulting from the allocation of C in Fig. 19). This step is skipped if the amount of remaining slots to be allocated for the data region is smaller than a full-height column.

3. The third step (32-36) allocates the remaining portion of the data region, which is by construction smaller than a full-height column (e.g., the third portion of C in Fig. 19). This step is skipped only if the number of slots of the data region before the second step is a multiple of H.

The detailed allocation procedure of H-ARQ data regions is described by means of the pseudo-code in Fig. 20. Data structures are initialized by creating an empty map (1) and a list of allocated data regions (2), and by setting the working variable width to zero (3). This variable will keep at each step the number of columns allocated so far. Note that there is no need to store the number of rows allocated so far, since the size of H-ARQ data regions is enforced to be a multiple of the downlink sub-frame height. The allocation procedure is performed by placing one of the data regions in `list_pending` at each iteration of the main loop (5). The iteration terminates when there are no more data regions to be allocated. The size of the DL-MAP is updated via the `add_harq_region()` function (8). As for the non-H-ARQ case, newly allocated data regions are added to the `list_allocated` data structures (12), which consists of a list of data region coordinates, expressed in terms of upper-left corner (i.e., starting column

```
1  // initialization
2  map = create_empty_map()
3  list_allocated = create_empty_list()
4  w = W
5  h = H
6  // loop until there are pending data regions
7  while ( ! list_pending.empty() )
8     // this is the data region under allocation
9     region = list_pending.pop()
10 // first step: allocate until the column is full
11    if ( h == H ) goto second_step
12    map.add_non_harq_region (region)
13    if ( W - map.width >= w ) break
14    size = min (h, region.slots)
15    region.slots -= size
16    h -= size
17    list_allocated.push (w, h, 1, size)
18    if ( h == 0 ) h = H, w--
19    if ( region.slots == 0 ) continue
20 second_step:
21 // second step: allocate multiple columns at once
22    if ( region.slots < H ) goto third_step
23    map.add_non_harq_region (1)
24    if ( map.width + w <= W ) break
25    cols = min (region.slots / H, map.width + w - W)
26    region.slots -= cols * H
27    w -= cols
28    list_allocated.push (w, 1, cols, H)
29    if ( region.slots == 0 ) continue
30 third_step:
31 // third step: allocate the remaining data
32    map.add_non_harq_region (1)
33    if ( W - map.width >= w ) break
34    h -= region.slots
35    if ( h == 0 ) h = H, w--
36    list_allocated.push (w, h, 1, region.slots)
```

Figure 29: Pseudo-code of SDRA without H-ARQ support.

and row, respectively, in slots) and dimensions (i.e., width and height, respectively, in slots). After the data region currently under allocation is extracted from the pending list (6), the number of columns needed to fully allocate it are computed and stored into `new_w` (7).

Then, there are two cases:

1. The data region fits into the remaining space, after the size of the DL-MAP has been updated (8). In this case the variable width is updated (11), a new data region is added to `list_allocated` (12), and the current data region is removed from the pending list (13). Allocation then restarts with the next data region in `list_pending` .

2. Otherwise, a sub-burst is removed from the current data region (18) and the allocation procedure restarts. If the sub-burst removed is the only one of the data region, then the latter is removed, as well (19). Note that the size of the DL-MAP must be also restored to its previous value (17) since no data region was allocated in this step.

To conclude this section we now compute the computational complexity of SDRA. With both non-H-ARQ and H-ARQ, the procedure is run as a single iteration, each of which only includes simple constant time operations (e.g., addition to/removal from the head of a list, elementary mathematical operations with integer values). Therefore, we can state that the worst-case computational complexity is $\mathcal{O}(n)$, where n is the number of data regions or the number of sub-bursts, respectively in the non-H-ARQ and H-ARQ cases.

Appendix B: Numerical instance generation

A numerical instance is generated as follows. Starting from an empty list, the input list of data regions is produced iteratively according to the following steps:

1. Initialize the offered load in slots (ol_S) and bytes (ol_B) to 0.

2. Consider a new SDU to be transmitted to a new user.

3. Determine the associated MCS mcs according to a probability distribution reflecting a typical MSs deployment scenario (Moi06).

```
map = create_empty_map()
list_allocated = create_empty_list()
width = 0
// loop until there are pending data regions
while ( ! list_pending.empty() )
   region = list_pending.top()
   new_w  =  region.slots / H
   map.add_harq_region (region)
   // the whole data region is allocated
   if ( width + map.width + new_w <= W )
      width += new_w
      list_allocated.push(W - width - new_w + 1, 1, new_w, H)
      list_pending.pop()
      /* the data region cannot be allocated as-is
         thus, a H-ARQ sub-burst is removed from
         the data region */
    else
      // remove region from the map
      map.remove_region (region)
      // remove a sub-item from region
      region.burst_list.pop()
      if ( region.burst_list.empty() ) list_pending.pop()
```

Figure 30: Pseudo-code of SDRA with H-ARQ support.

4. Determine the SDU type (i.e., application) according to the corresponding probability distribution: the SDU is VoIP with probability v, BE with probability $1 - v$.

5. Determine the number and size (in bytes) of PDUs used to transmit the SDU.

6. If VoIP was selected at point 3), then only one PDU is needed, of size s;

7. Otherwise, if BE was selected, the number n_{be} of PDUs is drawn from a geometric distribution with average $1/p_{be}$;

8. Determine the number and size (in slots) to transmit the above PDUs according to mcs. Each PDU is added to the data region reserved for MCS mcs. Moreover, in H-ARQ enabled snapshots, the PDUs are appended to the list of H-ARQ sub-bursts of data region mcs.

9. Update the offered load, both in bytes and slots. If ol_S ¡ tol restart from step 1.

10. As the final step, the set of data regions is sorted according to the ordering approach described above. There are three cases:

    - RND: Data regions are permuted according to a random uniform distribution.
    - MRF: Data regions are sorted deterministically in decreasing robustness order.
    - LRF: Data regions are sorted deterministically in increasing robustness order.

Chapter 6

Conclusions

> The press, the machine, the railway, the telegraph are premises whose thousand-year conclusion no one has yet dared to draw.
> *— Friedrich Nietzsche (1844 - 1900)*

In this work we presented PIPER, a modular framework to address the design of solutions for resource allocation in IEEE 802.16. The rationale of PIPER is to sub-divide the resource allocation process into separate functional sub-tasks. The latter are then carried out independently of each other and combined together in a pipeline manner so as to provide a complete solution for resource allocation. Specifically, we identified three general sub-tasks:

- *grant scheduling* provides connections with the negotiated QoS guarantees as specified by the IEEE 802.16 standard. To this end, it takes as input the connection QoS requirements and the current status of the connection queues and outputs a number of connection grants according to a QoS scheduling strategy;

- *grant arrangement* takes as input the output of the grant scheduling and re-arranges connection grants for i) reducing the system overhead and ii) enforcing some policies directly derived by the system constraints and specifications;

- *grant allocation* takes as input the output of the grant arrangement sub-task

and defines the final content of MAPs according to the specific air interface used.

We showed that the PIPER framework is very flexible for the following three reasons. First, different algorithms and approaches can be envisaged to address the same sub-task. For example, grant scheduling can be implemented picking one of the scheduling algorithms from the plethora available for wired networks, or different optimization strategies can be implemented as grant arrangement depending on the standard specifications. Second, our framework can apply to a wide variety of context. In this work, we actually showed a complete resource allocation solution for both IEEE 802.16d with OFDM and IEEE 802.16e with OFDMA exploiting the PIPER framework. Lastly, our approach might reduce the computational power required to perform resource allocation since the sub-tasks can be carried out concurrently.

Within the PIPER framework we developed two grant allocation algorithms, namely Half-Duplex Allocation (HDA) and Sample Data Region Allocation (SDRA), for IEEE 802.16d with OFDM and IEEE 802.16e with OFDMA, respectively. HDA can be employed by the BS for the grant allocation of SSs with both half- and full-duplex capabilities. By imposing a simple necessary condition to the grant arrangement sub-task, we proved that grant allocation of HD-SSs is always feasible and can be performed through HDA. The computational complexity of HDA is $\mathcal{O}(n)$, where n is the number of the grants to be allocated. We investigated the performance of HDA using an extensive simulation analysis, with VoIP and Web traffic, under varied load conditions and with different frame durations. Based on this analysis, we proved that, using HDA, the performance of SSs with half-duplex capabilities is almost equal to that of SSs with full-duplex capabilities. The performance degradation in uplink was due to the inability of HD-SSs to transmit while the BS is broadcasting the DL- and UL-MAPs.

Instead, SDRA can be adopted as grant allocator of rectangular data regions within the bi-dimensional OFDMA frame. We follow a heuristic approach as the grant allocation problem in OFDMA can be shown to be NP-hard since it is comparable to a bin-packing problem. SDRA has been designed so as to work with both non-H-ARQ and H-ARQ data regions. An extensive performance evaluation of SDRA has been carried out by means of Monte Carlo analysis under varied system and network configuration parameters, with both VoIP and BE users.

The following conclusions can be drawn from the results obtained. First, the smaller the MAC frame, the smaller the impact of the map overhead on the overall frame capacity, which anyway consumes a significant amount of the available bandwidth (i.e., up to 20%), and therefore the higher the probability that SDRA completely allocates the set of input data regions. Second, employing a reduced set of MCSs, i.e., three instead of the full set of seven MCSs, greatly reduces the map overhead, as well. However, there is a trade-off between the number of slots that are allocated per frame and the number of bytes that can be actually conveyed by them. In fact, lessening the set of available MCSs reduces the average transmission efficiency. Finally, the performance of SDRA has been shown to greatly depend on the order in which data regions are fed to the algorithm. For instance, ordering the data regions in increasing robustness order achieves the highest amount of data actually carried per frame, while ordering the data regions in decreasing robustness order produces the least amount of map overhead. While the same conclusions hold both for non-H-ARQ and H-ARQ data regions, the latter exacerbates the trade-off between transmission efficiency and map overhead.

References

[80204] IEEE 802.16. Air interface for fixed broadband wireless access systems, 2004. 3, 11

[80205] IEEE 802.16e. Air interface for fixed broadband wireless access systems, 2005. 3, 12

[BBT+07] L. Badia, A. Baiocchi, A. Todini, S. Merlin, S. Pupolin, A. Zanella, and M. Zorzi. On the impact of physical layer awareness on scheduling and resource allocation in broadband multicellular ieee 802.16 systems. *IEEE Wireless Commun.*, 14:36 – 43, February 2007. 5

[BCE+07] A. Bacioccola, C. Cicconetti, A. Erta, L. Lenzini, and E. Mingozzi. Bandwidth allocation with half-duplex stations in ieee 802.16 wireless networks. *IEEE Trans. Mobile Comput.*, 6:1384 – 1397, 2007. 45

[BCEM07] A. Bacioccola, C. Cicconetti, A. Erta, and E. Mingozzi. Throughput analysis of best effort traffic in IEEE 802.16/wimax. In *Proc. European Wireless*, Paris, France, April 2007. 5

[BEFN04] B. Bisla, R. Eline, and L. M. France-Neto. RF system and circuit challenges for WiMAX. *Intel Tech. J.*, 8:189–200, March 2004. 21

[Bou92] R. T. Boute. The euclidean definition of the functions div and mod. *ACM Transactions on Programming Languages and Systems*, 14(2):127–144, 1992. 35, 42

[Bra69] P. T. Brady. A model for generating on-off speech patterns in two-way conversation. *Bell System Technical J.*, 48:2445–2472, September 1969. 47

[Bro88] Robert Browning. *The Pied Piper of Hamelin*. Frederick Warne and Co., Ltd., 1888. 6

[BSKD06] Y. Ben-Shimol, I. Kitroser, and Y. Dinitz. Two-dimensional mapping for wireless ofdma systems. *IEEE Trans. on Broadcasting*, 52(13):388–396, September 2006. 60

[CELM06] C. Cicconetti, C. Eklund, L. Lenzini, and E. Mingozzi. Quality of service support in IEEE 802.16 networks. *IEEE Network*, 20:50–55, March 2006. 4, 11, 22, 29

[CELM07] C. Cicconetti, A. Erta, L. Lenzini, and E. Mingozzi. Performance evaluation of the ieee 802.16 mac for qos support. *IEEE Trans. Mobile Comput.*, 6:26–38, January 2007. 11, 14, 22, 29

[CH07] Y. Chen and F. Hsieh. A cross layer design for handoff in 802.16e network with ipv6 mobility. In *Proc. IEEE WCNC*, Kowloon, March 2007. 5

[Cim85] L. J. Cimini. Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing. *IEEE Trans. Commun.*, 33:665–675, June 1985. 14, 17

[CJW05] J. Chen, W. Jiao, and H. Wang. A service flow management strategy for IEEE 802.16 broadband wireless access systems in TDD mode. In *Proc. IEEE ICC*, pages 3422–3426, Seoul, Korea, May 2005. 27

[CLW+06] P. W. C. Chan, E. S. Lo, R. R. Wang, E. K. S. Au, V. K. N. Lau, R. S. Cheng, W. H. Mow, R. D. Murch, and K. B. Letaief. The evolution path of 4g networks: Fdd or tdd? *IEEE Communications*, 44(12):42–50, 2006. 21

[com] Redline communications. http://www.redlinecommunications.com/. 45

[CSR05] CM-SP-RFIv2.0-108-050408. Data-over-cable service interface specifications DOCSIS 2.0: Radio frequency interface specification. Technical report, Cable Television Laboratories, April 2005. 12

[EMP+06] C. Eklund, R. B. Marks, S. Ponnuswamy, K. L. Stanwood, and N. J.M. Van Waes. Wirelessman: Inside the IEEE 802.16 standard for wireless metropolitan area networks. *IEEE Press*, 2006. 67

[EMSW02] C. Eklund, R. B. Marks, K. L. Stanwood, and S. Wand. IEEE standard 802.16: A technical overview of the WirelessMAN air interface for broadband wireless access. *IEEE Commun. Mag.*, 40:98–107, June 2002. 16

[For] WiMAX Forum. http://www.wimaxforum.org/. 2

[For04a] WiMAX Forum. Business case models for fixed broadband wireless access based on WiMAX technology and the 802.16 standard. Technical report, 2004. 45

[For04b] WiMAX Forum. Initial certification profiles and the european regulatory framework. Technical report, 2004. 45

[GOS04] O. Gusak, N. Oliver, and K. Sohraby. Performance evaluation of the 802.16 medium access control layer. In *Proc. ISCIS*, pages 228–237, Kemer-Antalya, Turkey, October 2004. 29

[GWAC05] A. Ghosh, D. R. Wolter, J. G. Andrews, and R. Chen. Broadband wireless access with WiMAX/802.16: Current performance benchmarks and future potential. *IEEE Commun. Mag.*, 43:129–136, February 2005. 20

[HK06] S.-E. Hong and O.-H. Kwon. Considerations for VoIP services in IEEE 802.16 broadband and wireless access systems. In *Proc. IEEE VTC*, pages 1226–1230, Melbourne, Australia, May 2006. 12

[Hoy05] C. Hoymann. Analysis and performance evaluation of the OFDM-based metropolitan area network IEEE 802.16. *Computer Networks (Elsevier)*, 49:341–363, March 2005. 9, 16, 46

[HT06] H. Holma and A. Toskala. *HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications*. John Wiley & Sons, 2006. 2

[KC07] S. Ko and K. Chang. Capacity optimization of a 802.16e ofdma/tdd cellular system using the joint allocation of sub-channel and transmit power. In *Proc. Advanced Communication Technology*, February 2007. 5

[kid06] Cell phones for kids under 15: a responsible question. *Point.com*, 2006. 1

[KR04] I. Koffman and V. Roman. Broadband wireless access solutions based on OFDM access in IEEE 802.16. *IEEE Commun. Mag.*, 40:96–103, April 2004. 15

[KSC91] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis. Weighted round-robin cell multiplexing in a general-purpose ATM switch chip. *IEEE J. Select. Areas Commun.*, 9:1265–1279, October 1991. 28, 44

[LK00] A. M. Law and W. D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, 2000. 46, 66

[LL73] C. L. Liu and J. W. Layland. Scheduling algorithms for multi-programming in a hard real-time environment. *Journal of the ACM*, 20:46–61, January 1973. 27

[Moi06] J. Moilanen. OFDMA allocation numerical guidelines (confidential). Technical report, Nokia-Siemens Networks, July 2006. 81

[Mot00] Motorola. Evaluation methods for high speed downlink packet access. Technical Report TSGR#14(00)0909, 2000. 47

[MT90] S. Martello and P. Toth. *The Pied Piper of Hamelin*. John Wiley and Sons, 1990. 60

[NP00] R. Van Nee and R. Prasad. *OFDM for Wireless Multimedia Communications*. Artech House, 2000. 14, 17

[PG93] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services network. *IEEE/ACM Trans. Networking*, 1:344–357, June 1993. 29

[Pre01] Cisco Press. Traffic analysis for voice over IP. Technical report, 2001. 47

[SCGI05] S. Sengupta, M. Chatterjee, S. Ganguly, and R. Izmailov. Exploiting MAC flexibility in WiMAX for media streaming. In *Proc. IEEE WoWMoM*, pages 13–16, Taormina, Italy, June 2005. 9

[SV96] M. Shreedhar and G. Varghese. Efficient fair queueing using deficit round robin. *IEEE/ACM Trans. Networking*, 4:375–385, June 1996. 27, 44

[SV98] D. Stiliadis and A. Varma. Latency-rate serves: A general model for analysis of traffic scheduling algorithms. *IEEE/ACM Trans. Networking*, 6:611–624, October 1998. 27

[WG00] K. Wongthavarawat and A. Ganz. Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems. *Int. J. of Communication Systems (Wiley)*, 16:81–96, February 2000. 27

[WG03] K. Wongthavarawat and A. Ganz. IEEE 802.16 based last mile broadband wireless military networks with quality of service support. In *Proc. IEEE MILCOM*, pages 779–784, Boston, USA, October 2003. 27

[WH61] J. M. Wozencraft and M. Horstein. Coding for twoway channels. Technical Report 383, Massachussets Institute of Technology, 1961. 20

[Yag] Hassan Yaghoobi. Scalable OFDMA physical layer in ieee 802.16 wirelessman. Technical report. 17, 20

Index