Power-aware opportunistic downlink scheduling in IEEE 802.16 wireless networks

C. Cicconetti\textsuperscript{a,*}, L. Lenzini\textsuperscript{b}, D. Migliorini\textsuperscript{b}, E. Mingozzi\textsuperscript{b}

\textsuperscript{a} Dipartimento di Ingegneria dell'Informazione, University of Pisa, Via Diotisalvi 2, 56122 Pisa, Italy
\textsuperscript{b} Automotive and Telecommunications Division, Intecs S.p.A., Via E. Giannesi 5, 56121 Pisa, Italy

Abstract

The IEEE 802.16 is a standard for fixed and mobile Broadband Wireless Access (BWA). In this paper, we deal with two key challenges of 802.16-based networks. First, terminals close to cell edge experience poor channel quality, due to severe path-loss and high interference from concurrent transmissions in nearby cells. To address this issue, we propose a framework based on a static partitioning of bandwidth into chunks with different transmission power levels. Terminals with impaired channel conditions can then benefit from being allocated a higher amount of transmission power than the others. Secondly, transmissions should be scheduled according to Quality of Service (QoS) requirements to keep users with real-time video or voice calls satisfied, while best-effort connections should fairly share the remaining capacity. To this aim, we propose a scheduling algorithm, called Power-aware Opportunistic Downlink Scheduling (PODS), that aims at meeting both the QoS and fairness requirements, while taking into account the different power levels of the bandwidth chunks. The performance of the proposed scheduler is assessed through detailed packet-level simulation in realistic scenarios and compared with well-known scheduling algorithms. Results confirm that PODS is able to exploit power boosting to provide real-time connections with the desired level of QoS, irrespectively of their MSs' channel quality.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

The IEEE 802.16 standard [1] was originally designed for fixed Broadband Wireless Access (BWA), where Subscriber Stations (SSs) are deployed within the coverage range of a Base Station (BS) coordinating medium access in both downlink (from the BS to the SSs) and uplink (from the SSs to the BS). The air interface was initially based on either Single Carrier (SC) or Orthogonal Frequency Division Multiplexing (OFDM). More recently, the standard was enhanced to support Mobile Stations (MSs), and thus Orthogonal Frequency Division Multiple Access (OFDMA) was additionally specified as part of the air interface. While the research on IEEE 802.16 for fixed BWA is quite in a mature phase (e.g., [2]), mobility adds further challenges in the design of resource allocation algorithms and network deployment.

Among its features, the IEEE 802.16 specifies Quality of Service (QoS) support at the Medium Access Control (MAC) layer to enable interactive and multimedia applications along with traditional best-effort services. However, providing QoS guarantees is difficult, because the channel quality perceived by the MSs in the same cell can vary significantly over space and time, depending on environmental factors which cannot be controlled, e.g., the distance from the BS or moving obstacles. To receive the same amount of data, MSs with poor channel quality require more wireless resources than those with a good channel. Therefore, they either jeopardize the system's capacity or experience very low, possibly unbearable, QoS. For best-effort traffic only, this problem has been discussed, e.g., in [3], where the concept of temporal fairness is defined. Specifically, providing best-effort users with fair temporal sharing of the channel, rather than fair throughput, is proved to be more suitable in wireless networks.

In this paper, we face this problem by exploiting transmission power diversity, as supported by the standard, in a simple rather effective way. More specifically, we assume a frame structure where sub-carriers are statically allocated varied power levels, as allowed by IEEE 802.16. Based on this structure, we propose a Power-aware Opportunistic Downlink Scheduling (PODS) algorithm, whose objective is to provide real-time connections with a minimum transmission rate, whereas users with best-effort traffic are provided with fair temporal sharing of the channel, jointly exploiting both the channel and the power diversity. In particular, the MSs with real-time traffic are not given any special treatment, with respect to the best-effort, until their minimum rate is satisfied or they have good channel quality. This way, high throughput of best-effort traffic can be obtained without significant QoS degradation.

The rest of the paper is organized as follows. In Section 2 we describe the IEEE 802.16 MAC protocol, focusing on the structure of...
the OFDMA MAC frame. The proposed frame structure is discussed in Section 3. Then, in Section 4 we describe PODS, which is evaluated through simulation in Section 5. In Section 6 we discuss the related work, while conclusions are drawn in Section 6.

2. IEEE 802.16-OFDMA MAC/PHY

In this section we describe the aspects of the IEEE 802.16 MAC and physical (PHY) layer specifically relevant to this work. For a comprehensive description of the standard see [1]. As already mentioned, we only focus on the OFDMA air interface.

The IEEE 802.16 has a connection-oriented MAC protocol, where every uni-directional connection (CID) can be uniquely identified in the cell and has its own set of QoS parameters. The transmission of variable-length packets is supported by means of a convergence layer, which can also perform header suppression functions. One or more MAC Service Data Units (SDUs) can then be encapsulated into a single MAC Protocol Data Unit (PDUs). For efficiency purposes, MAC SDUs can also be fragmented. For Hybrid Automatic Repeat Request (H-ARQ) enabled connections, MAC PDUs are then concatenated into an H-ARQ sub-burst (sub-burst, for short), which is appended a Cyclic Redundancy Check (CRC) trailer, encoded and transmitted over the air. The correct/incorrect decoding of an H-ARQ sub-burst is indicated by the recipient MS by means of a dedicated logical sub-channel in uplink. Failed sub-bursts can be retransmitted by the BS up to a maximum number of times. The BS is allowed to have, for each MS, a limited number of H-ARQ sub-bursts pending the acknowledgement, in order to reduce the state associated to H-ARQ processing. Each H-ARQ process has associated an identifier called ACID.

Time is divided into fixed duration frames, each consisting of a number of time slots, whose duration is equal to one OFDM symbol. The spectrum is divided into equally-spaced sub-carriers. Sets of sub-carriers are logically combined into sub-channels, depending on the sub-channelization scheme employed. Two mandatory schemes are defined: Full-Usage Sub-Channelization (FUSC) and Partially-Used Sub-Channelization (PUSC). In any case, the minimum unit of transmission is called the slot, which consists of one or more time slots, in the time domain, and by one sub-channel, in the frequency domain. The exact duration and frequency width of a slot depends on the network configuration. Since the typical application scenarios of IEEE 802.16 involve the coexistence in the same cell of MSSs with very different channel conditions, a set of Modulation and Coding Schemes (MCSs) is defined in the standard. The more robust an MCS is, the smaller the number of bits that are conveyed per slot.

Transmission of sub-bursts occurs in data regions, which have a rectangular shape. In accordance with the standard, data regions can have an arbitrary size and position within the frame. These sizes and positions are advertised at the beginning of each frame in a MAC control message called Downlink Map (DL-MAP), which must be transmitted in the PUSC zone. Data addressed to different MSSs can be multiplexed into the same data region. The size and recipient of all sub-bursts are advertised in the DL-MAP, as well, along with other fields used for H-ARQ support, such as the retransmission index of the sub-burst and its sequence number. Since the DL-MAP is broadcast and is required by all the MSSs scheduled in a frame, the most robust MCS of those used in the frame must be used. A sample frame is illustrated in Fig. 1.

3. Proposed frame structure

Unlike fixed networks, where the relative position of the BS and SSs can be planned to some extent, multi-cell mobile networks are likely to suffer from time-varying highly-variable channel conditions for the various users. For instance, MSs moving near the cell edge will have severely degraded channel quality with respect to MSs that are closer to the BS, because of increased path loss and inter-cell interference. The traditional solution for mitigating this problem is to employ a fractional frequency re-use plan, where only a fraction of the sub-carriers are allocated to the BSs equipped with sectorized antennas.

A typical re-use fraction is 1/3. This way greatly reduces inter-cell interference, which yields better cell coverage. On the other hand, the cell capacity is reduced by the re-use factor. Hybrid approaches have been proposed to make a trade-off between capacity and coverage [4], where only a sub-set of the available sub-carriers uses fractional re-use.

Another opportunity that can be exploited in OFDMA wireless networks is downlink power control, which entails boosting the transmission power when transmitting to MSs with poor channel quality, either to enable them to receive, or to allow for more efficient MCSs to be employed [5]. Due to legal restrictions and to prevent network instability, the total power over all the sub-carriers is limited. Compared to frequency re-use, this approach is much more flexible, since boosting can be done in the time scale of frames. On the other hand, changing the re-use plan can involve complex operations, which may even require human interaction. However, exploiting boosting in IEEE 802.16 can become overly complex. In fact, as discussed in [6] in details, the problem of deciding the shape and position of data regions alone is a challenging and computationally-intensive task. While some heuristics have been put forward [14–17], research in this field is still in its infancy. Taking into account boosting brings the additional constraint that the power budget per OFDM symbol is limited, which further complicates the problem.

To break down the complexity, for the downlink sub-frame we adopt the structure illustrated in Fig. 2, where the shape of the all the data regions is the same: the duration is equal to the number of time slots in the FUSC zone, while the frequency span is equal to 1/L of bandwidth, where L is the number of sub-channels in a frequency re-use 1 plan. We call each data region a logical band. For every cell, extra power, with respect to the reference per-logical band amount, can be allocated to a sub-set of the logical bands. We call this assignment a boosting plan. The boosting plan is exploited by the scheduler described in the next section, which is the main contribution of this work.

Note that the static frame allocation illustrated in Fig. 2 also allows the area of the DL-MAP to be estimated in a more straightforward manner than with an arbitrary frame structure like that in Fig. 1. In fact, the DL-MAP size depends on the following factors: number of data regions, number of H-ARQ sub-bursts, and MCS of the MSSs with the poorest quality among those served in a frame. The number of data regions is constant and equal to L, while the other two factors can be estimated based on long-term measurements.
4. PODS scheduling algorithm

In this section we describe our proposed Power-aware Opportunistic Downlink Scheduling (PODS) algorithm. We assume that all MAC SDUs received from upper layers are enqueued as regular data into per-connection queues, both for real-time and best-effort traffics. MAC SDUs qualified as urgent can be promoted to: (a) H-ARQ sub-bursts to be re-transmitted, if they are scheduled and fail to be received by the recipient MS, or; (b) urgent data, if the connection experiences QoS depletion. The H-ARQ sub-bursts that need to be re-transmitted are held in per-MS buffers, since this is the default H-ARQ management of IEEE 802.16. However, urgent data remain in the regular MAC SDU queues until scheduled.

PODS algorithm is outlined in Fig. 3. At the beginning of every frame, scheduling is performed in three phases. Sub-bursts waiting to be re-transmitted are scheduled first. Then, the algorithm schedules any pending urgent data, from real-time connections. Finally, regular data are served. The frame can become exhausted in any of the three phases above, at which point scheduling terminates immediately. Due to the statistical multiplexing effects of real-time traffic flows and channel variations, it can happen that scheduling terminates during the second phase, i.e. regular data are not served at all, even for some consecutive frames. However, when this behavior is not transient but permanent, in other words only urgent data are served, QoS guarantees are not met anymore. To prevent such an undesired situation, Call Admission Control (CAC) should be performed by the BS during the establishment phase of any connection, which is supported by means of the set of standard messages of IEEE 802.16. The adaptation of existing CAC algorithms for IEEE 802.16 like, e.g., those proposed in [18,19], for a combined use with PODS is outside the scope of this document.

A detailed description of PODS is reported below, after the notations and assumptions have been introduced. An example of sample run of the proposed algorithm is reported in subparagraph C. Even though the scheduler was designed and evaluated with the IEEE 802.16 technology, its formulation is general enough to be applied to other OFDMA wireless systems. For this reason, we do not use the IEEE 802.16 terminology in the description, however the last part of this section has a discussion on the specific IEEE 802.16 implementation issues.

4.1. Notations and assumptions

Let us define $S$ as the number of slots per logical band, while $L$ is the number of logical bands. The total number of slots per frame is then $L \cdot S$. The queue of MAC SDUs of a given connection $i$ is indicated as $Q_i$. In the following we will use often the utility function $\sigma(Q_i, x, d)$, which returns the number of slots, limited by $d$ slots, that can be extracted from $Q_i$, when transmitting data with an MCS that conveys $x$ bits per slot.

With regard to the real-time connections only, two additional parameters are specified: $r$, which is the minimum reserved rate, in bytes/s, and $\tau$, which we call service interval, in seconds, which is the minimum interval over which the rate guarantee is met.

Finally, we assume that an Adaptive Modulation and Coding (AMC) function is implemented in the BS. More specifically, for a given connection $i$ the AMC function $\mu_i(b)$ returns the number of bits that can be conveyed per slot if data addressed to connection $i$ are transmitted in a logical band with boosting level $b$. If $\mu_i(b) = 0$ this means that connection $i$ cannot be allocated data in this frame in logical bands with boosting level $b$. Recall that the boosting levels $b_l$, $l \in \{1, \ldots, L\}$, for each logical band, are known to the scheduler from the boosting plan. In this work we do not consider the issue of determining $\mu_i(b)$, because the latter is used as input by the scheduler, along with the boosting plan. The notation used is summarized in Table 1.

4.2. Scheduling algorithm

We now describe in detail the three scheduling phases outlined above and illustrated in Fig. 3.

H-ARQ re-transmissions are served first in First-In-First-Out (FIFO) order. The rationale behind providing data in retransmission with the highest scheduling priority is that their connections have already undergone a selection process when the sub-bursts were scheduled originally for transmissions. If connections had to relinquish part of their fair share of the wireless channel for any H-ARQ re-transmission, the fairness would be degraded. Since we can reasonably assume that, in a steady state, the amount of data to be re-transmitted is a small fraction of the frame, in this phase we do not see the need for using a more sophisticated scheduling discipline than FIFO. If, however, the amount of data to be retransmitted increases (for example if a significant number of MSs experiences a very bad channel), could be necessary to use even entire frames to retransmit all the backlogged HARQ sub-burst. The logical band where the H-ARQ re-transmissions are allocated is selected based
on the same procedure described below for regular data, with straightforward changes that are not reported.

Let us now move onto the second scheduling phase. Every real-time connection $i$ has an urgent data counter $U_i$, in bytes, and a timer $T[i]$ associated to it. The event handlers and the urgent data scheduling function described above are formalized in Fig. 4 using a pseudo-code language. The timer is set to expire after $\tau_i$, whenever its state moves from inactive, i.e., $|Q_i| = 0$, to active, i.e., $|Q_i| > 0$. If the connection queue becomes empty before the urgent period elapses, the timer is canceled. In other words, data of real-time connection $i$ are treated as regular data while the queueing delay of every packet remains smaller than or equal to $\tau_i$. When this condition fails, the timer $T[i]$ expires. At this point, the enqued data become urgent: the urgent counter $U_i$ is incremented by $r_i = \tau_i$, the timer $T[i]$ is re-started with a value of $\tau_i$, and, the connection identifier is inserted at the tail of a FIFO list, called urgent list. Note that the urgent counter is not allowed to overflow the current amount of backlog. Otherwise, a connection would be allowed to gain priority for data that has not arrived yet, which is unfair.

In accordance with the urgent list, the connections with urgent data are scheduled up to $U_i$ bytes in the same order in which their timers expired. The logical band where the urgent data are allocated is selected by the function $selectLogicalBand$, which is described below. The function $scheduleData(i, l, U_i)$ creates a H-ARQ sub-burst to be allocated into the logical band $l$, containing as many MAC SDUs from $Q_i$ (of fragments thereof) as possible provided that their total size fits into the available room in the logical band $l$ and it does not exceed $U_i$. The H-ARQ sub-burst size, in bytes, is returned by the function, and it is used to update the urgent counter $U_i$. As for the allocation of sub-bursts to be re-transmitted, this function can be trivially derived from the allocation procedure for regular data, which is described in the following.

We conclude the description of the proposed algorithm by illustrating the procedure for scheduling regular data, which is done in a round-robin manner. Regardless of its type, any connection $i$ is then associated with a quantum $q_i$, which is the number of slots that connection $i$ is granted every time it is visited by the scheduler while active. The amount of service that cannot be exploited in the current visit is kept in a status variable called credit $(\delta_i)$, inspired by [7]. For all inactive MSs it is $\delta_i = 0$. The reason for keeping the quantum and credit in slots, rather than bytes, is that the slot is a constant fraction of the wireless channel, irrespectively of the quality of the MSs to which data are addressed. This means that temporal fairness can be provided, thus meeting the scheduler's objective. Furthermore, by assigning different quanta $q_i$'s to connections, it is possible to provide weighted fair service. Given two non-real-time connections $i$ and $j$ with quantum $q_i$ and $q_j$, respectively, if they are both active for a prolonged time window, the ratio between the number of slots scheduled to connection $i$ and to connection $j$ will be $q_i/q_j$. More specifically, the quantum of connection $i$ is:

$$q_i = \frac{\phi_i}{\sum_{other\ connections} \phi_j} \Theta,$$

(1)

where $\phi_i$ is the weight of connection $i$ and $\Theta$ is the target round duration, in slots, which is a system parameter discussed at the end of the current section.

For efficiency reasons, an active list is kept that contains the identifiers of the active connections only. The identifier of a connection is added to the tail of the active list when its state changes from inactive to active, while it is removed when the connection becomes inactive.

We now describe the $selectLogicalBand$ function, which selects the logical band for transmitting data when any connection $i$ is visited by the round-robin algorithm for non-urgent data. The current residual of logical band $l$ is $a_i$, $l \in \{1, \ldots, L\}$, which is the number of slots still available for scheduling new data. We define the feasible set of connection $i$, $\Phi_i$, as the set of logical bands that can be used for transmitting data of connection $i$:

$$\Phi_i = \{l | \mu_i(b_l) > 0 \land a_l > 0\}.$$

(2)

The logical band selected for transmission, $l \in \Phi_i$, is that one which leaves the smallest residual, among those that can occupy as many slots as possible in the logical band, with the minimum boosting level. To select $l$ we perform three incremental reductions of the set $\Phi_i$.

First, we remove from $\Phi_i$ all the logical bands that do not have the minimum boosting level:

$$\Phi'_i = \{l \in \Phi_i | b_l = \min_{k \in \Phi_i} \{b_k\}\}.$$

(3)

This step is carried out to give those connections that would not be scheduled without extra power allocation a higher chance to transmit, i.e. any connection $j$ for which $\mu_j(0) = 0$. Then, we remove all the logical bands that do not occupy the maximum amount of slots:

$$\Phi''_i = \{l \in \Phi'_i | s_l = \sigma(Q_i, \mu_i(b_l), \delta_i)\}.$$

(4)

where $s_l$ is the number of slots that can be allocated to connection $i$ in logical band $l : s_l = \sigma(Q_i, \mu_i(b_l), \delta_i)$. The reason for selecting the logical bands with the greatest amount of free slots is that the
MAC overhead can be reduced by exploiting the packing and concatenation mechanisms of the IEEE 802.16 MAC. A study on how to optimize the H-ARQ sub-burst size as a function of the system parameters can be found in [8]. Finally, we select \( l \) as:

\[
\hat{l} = \arg \min_{l} \{ a_l - s_l \}.
\]  

Before actually scheduling data for connection \( i \), we update the credit of connection \( i \) as follows (only for regular data):

\[
\delta_n = \delta_n + q_n.
\]

If the credit is not enough to fully occupy the selected logical band or no logical band can be selected, i.e.:

\[
\delta_n < s_l \quad \text{or} \quad \Phi^u = \emptyset,
\]

then the connection does not receive service in this round. This means that the round-robin pointer is moved to the next connection in the active list, while the connection in an increased credit that it will be able to spend in the next visits, in the current or subsequent frames.

We conclude the description discussing the problem of selecting appropriate values for the quantas, or \( \Theta \), which is common to round-robin schedulers and has been studied extensively in the literature of packet scheduling algorithms for Internet routers [9]. As a matter of fact, any values could be used in principle, so as to achieve the scheduling objective of providing weighted fairness. However, using too small \( q_i \)'s increases the time required to perform scheduling, since the scheduler might be forced to perform several visits before scheduling any connection due to condition (7). On the other hand, using too large \( q_i \)'s can lead to a bursty service, since any connection will be given service for at least \( q_i \) slots at every visit, possibly “capturing” the channel for long periods and, hence, increasing the overall queuing delays. As a rule of thumb, in PODS we select \( \Theta \) such that the smallest quantum is equal to the logical band size, in slots.

4.3. Scheduling example

To better illustrate the scheduling algorithm described in the previous section we use the following example. Let us assume the scheduler is visiting connection \( i \). The current status of the residual in the logical bands is depicted in left-most part of Fig. 5, with \( L = 8 \), and boosting plan \( \{3,0,6,3,0,0,3,0\} \) in dB.

Assume that the MS of connection \( i \) requires at least a boosting level of 3 dB, i.e. \( \mu_i(0) = 0 \). Thus, the feasible set \( \Phi_i \) only includes logical bands 1, 3, 5 and 7, because the others do not have a high enough boosting level. The logical band 3 is then removed when reducing \( \Phi_i \) to \( \Phi_i' \), because its boosting level is not minimum. Let us consider now four possible backlog values, in slots, for connection \( i \), as reported in Fig. 5: 1, 3, 5 and 6. For simplicity of illustration, we assume that each packet occupies exactly one slot, with no fragmentation/packeting overhead.

With a single slot backlog, all the logical bands \( \Phi_i' \) can fit the entire amount of data buffered at connection \( i \). Thus, \( \Phi_i'' = \{1,4,7\} \) because the number of slots occupied for all bands is the same, and is equal to 1. With a backlog of 3 slots, instead, the logical band 7 is not included in \( \Phi_i'' \) because it can hold at most 2 slots, unlike the others. Finally, if there are 5 or 6 slots of backlogged data, then only the logical band 4 is included in \( \Phi_i'' \). When there is more than one logical band in \( \Phi_i'' \), the tie is broken based on the residual that would result for each band if it were selected for data transmission.

The final logical bands selected for transmission are reported in the right-most part of Fig. 5: logical band 7 for the case of 1 slot backlog; logical band 1 for the case of 3 slots backlog; logical band 4 in the other cases, with backlog equal to 5 or 6 slots.

4.4. Implementation issues

The definition of QoS in terms of \( r_i \) and \( \tau_i \) is flexible enough to suit a large number of applications. For instance, interactive applications, such as packetized voice or video calls, will likely require a small value of \( \tau_i \) in the order of tens of milliseconds, to avoid out-of-sync playback, with \( r_i \) equal to the application peak rate. On the other hand, real-time applications with stringest delay requirements, such as buffered video streaming, can specify a value of \( r_i \) large enough not to cause overflow of the playback buffer, e.g. in the order of seconds, and \( r_i \) equal to the encoder average rate.

With specific reference to IEEE 802.16, every connection belongs to one of the following five scheduling services. The Unsolicited Grant Service (UGS) and extended real-time Polling Service (eRTS) scheduling services are mainly intended for uplink connections; hence we do not consider them here. The real-time Polling Service (RTS) scheduling service is designed for use with interactive applications with stringent delay requirements. For this kind of connections, the Maximum Sustained Traffic Rate and Maximum Latency parameters can be used to set the value of \( r_i \) and \( \tau_i \), respectively. On the other hand, non-real-time Polling Service (nRTS) connections are intended for non-interactive applications, with large bandwidth requirements. In this case, the Minimum Reserved Traffic Rate and Maximum Traffic Burst can be used to set the value of \( r_i \) and \( r_i - \tau_i \), respectively. Finally, the best-effort (BE) scheduling service is for connections with no specific QoS requirements, which suits well the best-effort class of our proposed algorithm.

5. Performance evaluation

In this section we report the simulation analysis on the performance of PODS. The simulations were carried out using a dynamic system level simulator. The network configuration we used was
Based on the WiMAX Forum specifications [10]. The values of the parameters that are most relevant for this study are reported in Table 2.

5.1. Channel model

Since the objective of this study is to analyze the properties of PODS and study how it behaves compared to known schedulers from the literature, simplifying approximations were introduced when modeling the channel for our simulations. Furthermore, we simulate a single cell and we assume constant co-channel interference from nearby cells, in order to focus on the properties of the scheduling algorithm alone. Investigations with more sophisticated channel models, also including time-varying co-channel interference, are left as future work.

We now explain how the propagation channel was modeled. Different channel instances were generated, one per MS per simulation replication. For each instance, the time-correlated channel coefficients were then aggregated to produce a channel trace of Signal-to-Noise-Ratio (SNR) samples, one per frame [20]. Distance-based path-loss and lognormal shadow fading were then applied, whereas inter-cell interference and antenna gains were not considered. In the simulations, the correct reception of a sub-burst is a function of its size and Block Error Rate (BLER). The latter, in turn, depends on the MCS used for transmission and the SNR of the frame when the sub-burst is received, as well as the SNR of any previous transmission of the same sub-burst. Such mapping was derived from the specifications in [21] for Additive White Gaussian Noise (AWGN) channels, which is known to yield optimistic performance compared to fading channels. The channel model parameters are reported in Table 2. Any MS is assigned one of the six combinations of SNR level, MS speed, and MS distance in Table 2 in a deterministic uniform manner.

MSs estimate their average SNR by applying an EWMA filter to the per-frame SNR samples read from the channel trace, with a smoothing factor equal to 0.01. The current SNR value of an MS is then used to report the desired MCS to the BS every Channel Quality Indication (CQI) reporting interval. We assume that CQI reports are always decoded correctly by the BS. The reported MCS is the most efficient to achieve a probability smaller than 0.005 that a 480 bits block is incorrectly received, without H-ARQ recovery. Based on the last CQI message received from any MS i, the BS then selects the MCS used for transmission, depending on the boosting level b, according to function $\mu_i(b)$ defined in Section 4. In other words, every MCS is re-mapped to another with the same or higher efficiency, depending on the amount of extra power granted. To evaluate their relative impact on the performance, simulations were run with two different re-mapping functions. These functions, reported in Table 3, are referred to as conservative and aggressive, respectively, and have been derived empirically. Note that the aggressive remapping yields a use of higher order modulations in the boosted logical bands than the conservative remapping. This increases the transmission efficiency, but increases the probability that transmitted data cannot be decoded by the receiver because of channel errors, which, in turn, increases the average number of re-transmissions per H-ARQ sub-burst.

5.2. Traffic models

In the analysis we used three traffic types: Voice over IP (VoIP), Video on Demand (VoD), and data. VoIP traffic is modeled as an ON/OFF process, with silence and talkspurt periods distributed according to Weibull distributions. The packet size and generation interval are selected according to the GSM AMR specifications, which is one of the most employed codecs in wireless cellular networks. Robust header compression at the MAC layer is assumed, with only 3 bytes needed to convey RTP/UDP/IP data [1]. A VoD source consists of a traffic trace from a pre-encoded MPEG4 file [11], with randomized starting offset. Finally, data traffic is modeled as an uninterrupted source of fixed-size packets to emulate asymptotic conditions, so as to derive the capacity limits of networks and to produce maximum interference to real-time traffic. The configuration values of the traffic models are reported in Table 4. In PODS, all the connections are assigned the same weight parameter because of channel errors, which, in turn, increases the average number of re-transmissions per H-ARQ sub-burst.

5.3. Comparison algorithms

PODS was tested with and without the urgent list mechanism enabled, and with both the re-mapping functions in Table 3. When

### Table 2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channel specifications (SNR traces only)</td>
<td>ITU Pedestrian-A (Ped-A), ITU Vehicular-A (Veh-A)</td>
</tr>
<tr>
<td>MS distance</td>
<td>300 m, 600 m, 900 m</td>
</tr>
<tr>
<td>MS speed</td>
<td>1 m/s (Ped-A), 20 m/s (Veh-A)</td>
</tr>
<tr>
<td>BS's transmission power</td>
<td>30 dBm</td>
</tr>
<tr>
<td>Thermal noise level</td>
<td>–10 dBm</td>
</tr>
<tr>
<td>Shadowing std. deviation</td>
<td>8 dB</td>
</tr>
<tr>
<td>Path-loss exponent</td>
<td>3.5</td>
</tr>
<tr>
<td>CQI type</td>
<td>effective CNIR</td>
</tr>
<tr>
<td>CQI reporting interval</td>
<td>2 frames</td>
</tr>
<tr>
<td>MCS reporting target BLER</td>
<td>0.005 (1 block = 480 bits)</td>
</tr>
<tr>
<td>Duplexing mode</td>
<td>Time Division Duplexing</td>
</tr>
<tr>
<td>OFDM symbol duration</td>
<td>102.86 μs</td>
</tr>
<tr>
<td>Frame size</td>
<td>5 ms (47 OFDM symbols)</td>
</tr>
<tr>
<td>No. downlink sub-channels</td>
<td>30 (PUSC), 16 (FUSC)</td>
</tr>
<tr>
<td>No. logical bands (FUSC only)</td>
<td>8 (2 sub-channels each)</td>
</tr>
<tr>
<td>Downlink FUSC zone (control)</td>
<td>7 OFDM symbols</td>
</tr>
<tr>
<td>Downlink FUSC zone (data)</td>
<td>28 OFDM symbols</td>
</tr>
<tr>
<td>Max No. H-ARQ re-transmissions</td>
<td>4</td>
</tr>
<tr>
<td>H-ARQ DL burst delay</td>
<td>1 frame</td>
</tr>
</tbody>
</table>

### Table 3

<table>
<thead>
<tr>
<th>MCS</th>
<th>0 dB</th>
<th>Conservative $\mu_i$</th>
<th>Aggressive $\mu_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3 dB</td>
<td>6 dB</td>
<td>9 dB</td>
</tr>
<tr>
<td>QPSK-1/2 rep. 6</td>
<td>8</td>
<td>0</td>
<td>48</td>
</tr>
<tr>
<td>QPSK-1/2 rep. 4</td>
<td>12</td>
<td>0</td>
<td>48</td>
</tr>
<tr>
<td>QPSK-1/2 rep. 2</td>
<td>24</td>
<td>48</td>
<td>72</td>
</tr>
<tr>
<td>QPSK-1/2</td>
<td>48</td>
<td>48</td>
<td>72</td>
</tr>
<tr>
<td>QPSK-3/4</td>
<td>72</td>
<td>72</td>
<td>96</td>
</tr>
<tr>
<td>16-QAM-1/2</td>
<td>96</td>
<td>96</td>
<td>144</td>
</tr>
<tr>
<td>16-QAM-3/4</td>
<td>144</td>
<td>144</td>
<td>144</td>
</tr>
<tr>
<td>64-QAM-1/2</td>
<td>144</td>
<td>144</td>
<td>192</td>
</tr>
<tr>
<td>64-QAM-2/3</td>
<td>192</td>
<td>192</td>
<td>216</td>
</tr>
<tr>
<td>64-QAM-3/4</td>
<td>216</td>
<td>216</td>
<td>240</td>
</tr>
<tr>
<td>64-QAM-5/6</td>
<td>240</td>
<td>240</td>
<td>240</td>
</tr>
</tbody>
</table>

### Table 4

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VoIP Rate (during talkspurts)</td>
<td>12.8 kbps</td>
</tr>
<tr>
<td>Talkspurt period (Weibull)</td>
<td>$\lambda = 1.423, \kappa = 0.824$</td>
</tr>
<tr>
<td>Silence period (Weibull)</td>
<td>$\lambda = 0.899, \kappa = 1.089$</td>
</tr>
<tr>
<td>Urgent parameters: $\tau, r$</td>
<td>100 ms, 12.8 kbps</td>
</tr>
<tr>
<td>VoD MPEG4 trace</td>
<td>Futurama (medium quality)</td>
</tr>
<tr>
<td>Mean (peak) rate</td>
<td>0.31 (5.7) Mb/s</td>
</tr>
<tr>
<td>Urgent parameters: $\tau, r$</td>
<td>2 x 310 kbps</td>
</tr>
<tr>
<td>Buffer size</td>
<td>500 kbps</td>
</tr>
<tr>
<td>Data Packet size</td>
<td>100 byte</td>
</tr>
<tr>
<td>Rate</td>
<td>Infinite</td>
</tr>
<tr>
<td>Urgent parameters: $\tau, r$</td>
<td>0 s, 0 kbps</td>
</tr>
</tbody>
</table>
unspecified, aggressive re-mapping was used. In addition to PODS, the following well-known scheduling algorithms were implemented for comparison purposes: Max-C/I and Proportional Fair (PF). Max-C/I strives to maximize the overall channel utilization, by scheduling data in decreasing order of transmission efficiency. PF, instead, was originally designed in [12] to provide users with long-term fairness, in terms of their throughput, by exploiting short-term channel diversity. Finally, we employed a scheduling policy, called PrioPF, where the real-time traffic is served with strict priority with respect to best-effort data. In the scenarios simulated, the total load of real-time traffic never exceeds the system capacity, which emulates the existence of a CAC algorithm. Therefore, under any scheduling policy, the transmission of real-time traffic is stable, i.e. there is no packet loss due to buffer overflow. The remaining capacity after all real-time traffic has been scheduled is shared by best-effort connections using PF. While advanced scheduling algorithms, exploiting specific traffic and environment characteristics, have been put forward in the literature, see [6] and references therein, we chose Max-C/I, PF, and PrioPF because they have known properties that hold in general and, hence, they provide reference results useful for the analysis of PODS.

To perform a fair comparison, Max-C/I, PF, and PrioPF were adapted to exploit the same frame structure as PODS. For each logical band, the active connections are sorted according to a scheduler-dependent ranking criterion. The top connection is then served until either there are no any more data backlogged or the logical band is full. The next logical band is then considered, and so on until the whole zone is full. With both schedulers ties are broken randomly at each logical band. With regard to the ranking criterion, Max-C/I selects the connection whose MS has the most efficient MCS, while PF selects the connection $i$ such that:

$$i = \arg \max_j \{r_j/R_j\},$$

where $r_j$ is the instantaneous transmission rate of connection $j$, and $R_j$ is its long-term rate, which is estimated through an Exponentially Weighted Moving Average (EWMA) filter:

$$R_i = (1 - \alpha)R_i + \alpha \cdot r_j$$

with $\alpha \in (0,1)$. The long-term rate was updated once every frame.

5.4. Simulation results

Every scenario was simulated separately for each scheduler, and with 0, 3, and 6 boosted logical bands. In the case with 3 logical bands, the boosting levels used were 3 dB, 6 dB, and 9 dB. With 6 logical bands, the same boosting level was repeated twice. For each combination of parameters we ran 20 independent replications, whose samples were averaged to obtain an estimated mean and 95% confidence interval, which is reported in the figures, unless negligible. The replications of the same simulation differed in the following characteristics: VoIP and VoD traffic pattern; position of the boosted logical bands; channel quality, though the combination of speed and distance of any MS remains the same. Several cell loads and mixes of MSs were simulated, whose results are discussed in the rest of this section. For reasons of space, only a selection of the most relevant results obtained are reported. In the first scenario the following traffic mix was used: 30 VoIP, 10 VoD, and 20 data connections, one per MS. The results are reported in Fig. 6. We analyze the results for each traffic type separately.

For VoIP traffic, we used the Mean Opinion Score (MOS), between 1 (unbearable quality) and 5 (best quality), as the performance index [13], which combines together the delay and packet loss of VoIP frames. We then assume that VoIP users will be satisfied if 75% of their talkspurts obtain a MOS greater than or equal to 3.5. Finally, the VoIP satisfaction is the ratio of the satisfied users to the total number of users in the cell (Fig. 6). As can be seen, with
Max-C/I none of the VoIP users are ever satisfied, because their traffic is served only when they have good channel conditions, otherwise full-buffer MSs saturate the channel. Like Max-C/I, PF does not differentiate among traffic types, but it is able to cope well with VoIP traffic, hence achieving the same performance level as the Prio (Max-C/I, PF) scheduler. This is because PF aims at providing all connections with the same rate. Since VoIP connections have very low rate requirements, such provisioning is enough for them to obtain a timely service. Finally, PODS, with urgent list enabled, has the same VoIP outage as PF and Prio (Max-C/I, PF) without boosting, but performs better when there are some boosted logical bands due to its wise data allocation and MCS re-mapping. In fact, the QoS degrades significantly when the urgent list is disabled.

We now analyze the performance of VoD traffic. Since VoD applications have less stringent delay requirements than VoIP, performance is assessed in terms of the packet loss only, as reported in Fig. 6 separately for every VoD connection. The identifiers of VoD connections on the x-axis have been sorted in decreasing channel quality of their MSs, in terms of their average SNR sampled. Only the case with no boosting is reported, since the results with 3 and 6 boosted logical bands are similar. Both Max-C/I and PF yield a high packet loss for those MSs with poor average channel quality, i.e. from 5 to 10 for PF, and from 9 to 10 for Max-C/I. This is because data connections have infinite demand, and thus compete aggressively for bandwidth sharing with VoD connections, which have much greater rate requirements than VoIP. In fact, the only scheduler that keeps pace with PODS is Prio (Max-C/I, PF). This is because data connections have infinite demand, and thus compete aggressively for bandwidth sharing with VoD connections, which have much greater rate requirements than VoIP. In fact, the only scheduler that keeps pace with PODS is Prio (Max-C/I, PF). As with VoIP, disabling the urgent list mechanism in PODS leads to poor performance, especially for those MSs that have poor average channel quality.

The Prio (Max-C/I, PF) scheduler is the only valid alternative to PODS for serving QoS traffic with not significant degradation.

However, it incurs severe penalties with respect to PODS in terms of the cell throughput, shown in Fig. 6. This is especially true with 3 and 6 boosted logical bands. While Max-C/I outperforms all the other schedulers, PODS achieves a higher throughput than PF too, except when no logical bands are boosted. Moreover, the absence of an urgent list penalizes QoS traffic, but does not improve the cell throughput significantly. Finally, in all the results presented for this scenario, the boosting plan with 6 boosted logical bands does not produce significantly better results than those with only 3 boosted logical bands.

In Fig. 7 we report the results obtained when the cell workload consists of a constant number of VoIP and VoD connections, i.e. 6 each, and the number of data connections increases from 0 up to 90. Only the results with 3 boosted logical bands are reported. The VoD packet loss is reported only for the MS with worst average channel conditions. The objective is to evaluate the isolation of real-time traffic, with increasing disturbance from best-effort MSs. As expected, Max-C/I and PF, which do not differentiate services, perform badly, even with a relatively small number of data MSs. Prio (Max-C/I, PF), on the other hand, is very resilient to the injection of data traffic, because it gives real-time connections strict priority. However, PODS is able to get the best of both worlds, by providing a high level of QoS, while increasing the cell throughput, with respect to Prio (Max-C/I, PF). Finally, the advantage of the urgent list in PODS outlined in the previous scenario is confirmed in this scenario as well.

We now analyze the behavior of PODS in the same scenario, by comparing the results obtained with the two re-mapping functions defined in Table 3. First, we report in Fig. 8 the probability that a sub-burst is not received correctly by the recipient MS the first time it is transmitted. With both boosting plans tested, the curve obtained with an aggressive MCS re-mapping leads to a higher transmission rate, for all MSs. The case with no boosted logical
bands is not reported because there is no difference between the two cases considered. Aggressively using higher MCSs in boosted logical bands, thus, leads to an increased number of H-ARQ re-transmissions, which consume the channel capacity.

However, the capacity wastage due to re-transmitting more often data is compensated for by the higher transmission efficiency, defined as the ratio between the total number of bits correctly received by a connection, and the total number of slots that have been occupied by it. Fig. 9 shows the ratio between the transmission efficiency with the aggressive MCS re-mapping and that with conservative re-mapping. As can be seen, the curves with both boosting plans are above one, which explains why the application-related measures for the two cases are approximately the same.

We conclude our analysis with a scenario with 24 MSs with data traffic only, specifically designed to show the fairness properties of PODS. After a simulation warm-up period, we extracted the time-trace of the amount of service received per frame by each connection. We then processed the data to obtain, for different time intervals, the average service difference, which is the difference between the number of slots scheduled to the most served and least served connections. The results obtained with PODS, PF, and Max-C/I, with 0 and 3 boosted logical bands, are reported in Fig. 10. Two issues are especially worth noting. First, PA significantly improves the fairness in the short term, i.e. for time windows smaller than 100 ms. Second, unlike with PF and Max-C/I, when the measurement interval grows larger, the average service difference of PA is bounded, hence long-term fairness is ensured.

6. Related work

In the last decade several scheduling algorithms have been proposed and analyzed in the context of WiMAX networks. A comprehensive survey of the most recently proposed ones can be found in [6].

In general, scheduling algorithms for WiMAX can be arranged into two main classes: channel unaware vs. channel aware. Schedulers belonging to the former class do not care about information on the radio channel quality experienced by the MSs. On the contrary, channel aware algorithms use in the scheduling decision the reports on the channel quality provided by the MSs.

Channel-aware schedulers can be further classified depending on their key objective, which can be one of (i) ensuring fairness, (ii) providing QoS guarantees, (iii) maximizing system throughput, and (iv) optimizing transmission power, or a combination of them.

In [24,25,34] the main goal is to ensure fairness without necessarily providing QoS guarantees. Wengerter et al. [25] propose a Generalized Proportional Fair (GPF) scheduling algorithm in which they define two different fairness indexes (Allocation and Data-rata Fairness, respectively) to evaluate the scheduler. In [34] Bonnald investigates the limits of the well-known Proportional Fair (PF) scheduler [12], and proposes a score-based opportunistic scheduler to deal with this issues.

Enhancements to the PF scheduler are further proposed in [22,23,31,32] in order to support QoS. In particular, Ryu et al. [23] propose an algorithm, called UEPS, that uses a time-utility function as a scheduling urgency factor jointly with the status of the current channel as an indicator of the radio resource usage. Hou et al. propose in [31] an algorithm to dynamically tune the parameters of the PF algorithm: by using different time windows to estimate the throughput they show that it is possible to differentiate the queuing delay. On the other hand, Khattab and Elsayed [32] split the scheduling decision into two sub-problems: the OFDMA subcarrier allocation followed by the subcarrier assignment. Won-Hyoung et al. [29] propose a scheduler for multiple traffic classes that is an extension of Modified Largest Weighted Delay First (M-LWDF) scheduler originally proposed in [30]. In [33] Puzzi et al. compare compensation-based and greedy approaches according to their ability to provide QoS and fairness to power traffic flows.

While having more boosted logical bands degrades the fairness achieved by PA, the service difference is nevertheless bounded in the long run.
a cross-layer optimization algorithm which takes into account the adaptive modulation and coding, the allocation of time–frequency resources and the transmit power on each frequency sub-channels. The corresponding scheduler, however, does not consider fairness. The scheduler in [28] is proposed by Zhang et al. and uses water filling as a power allocation algorithm based on PF. QoS is guaranteed in terms of minimum rates, but without considering any delay constraint. Moreover, the water filling algorithm provides only a small improvement compared to the fixed power allocation with adaptive modulation and coding as demonstrated in [35].

Finally, modern cellular networks like WiMAX are afflicted by the cell-edge user problem, which is investigated in [36–38]. The solutions proposed in these works exploit the cooperation among adjacent cells of the network, which however increases significantly the scheduler complexity and overhead.

The scheduler proposed in this work, unlike the schedulers cited so far, takes into consideration all the following design objectives at the same time: QoS, fairness, power allocation, and the cell-edge user problem. In particular, to solve the problem of cell-edge users a static power allocation is adopted. Such a solution does not increase the complexity of the scheduler and does not need any type of cooperation among cells. Finally, thanks to the urgent list it is possible to guarantee a minimum rate or/and a maximum delay while providing at the same time an acceptable fairness.

7. Conclusions

In this paper we have proposed a downlink scheduling algorithm for mobile users in IEEE 802.16 wireless networks, called PODS. The algorithm is aimed at serving the MSs efficiently according to their channel quality, while guaranteeing a minimum rate to QoS connections. Our scheduler exploits a frame structure where some logical bands are provided with extra transmission power.

We have compared the performance of PODS to that obtained with popular scheduling algorithms for wireless networks, i.e. Max-C/I and PF, with and without prioritization of real-time traffic. The results, which were obtained through detailed packet-level simulation in a single-cell network, show that the proposed scheduling algorithm is able to boost efficiency sufficiently to give high throughput. At the same time, it provides VoIP and VoD real-time connections with the desired level of QoS, irrespective of their MSs’ channel quality. In addition, we have verified that it provides best-effort users with temporal fair usage of the wireless resources, both in the short and long term.

References