14

Resource Management Framework for QoS Scheduling in IEEE 802.16 WiMAX Networks

Hua Wang and Lars Dittmann
Networks Technology and Service Platforms
Department of Photonics Engineering
Technical University of Denmark, Lyngby, Denmark
hwan@fotonik.dtu.dk, ladit@fotonik.dtu.dk

Abstract
IEEE 802.16, also known as WiMAX, has received much attention recently for its capability to support multiple types of applications with diverse Quality-of-Service (QoS) requirements. Beyond what the standard has defined, radio resource management (RRM) still remains an open issue, which plays an important role in QoS provisioning for different types of services. In this chapter, we propose a downlink resource management framework for QoS scheduling in OFDMA based WiMAX systems. Our framework consists of a dynamic resource allocation (DRA) module and a connection admission control (CAC) module. A two-level hierarchical scheduler is developed for the DRA module, which can provide more organized service differentiation among different service classes, and a measurement-based connection admission control strategy is introduced for the CAC module. Through system-level simulation, it is shown that the proposed framework can work adaptively and efficiently to improve the system performance in terms of high spectral efficiency and low outage probability.

Keywords: WiMAX OFDMA radio resource management QoS scheduling

1. Introduction

Over the last decade, the rapid growth of high-speed multimedia services for residential and small business customers has created explosive demand for last mile broadband access. Currently, most broadband access is offered through wired lines, such as xDSL, cable or T1 networks. However, there are still a large number of areas where wired infrastructures are difficult to be deployed because of technical or commercial reasons. Broadband Wireless Access (BWA) systems are gaining extensive interests from both industry and research communities due to the advantages of rapid deployment, lower maintenance and upgrade costs, and granular investment to match market growth (1). Among the emerging technologies for BWA, IEEE 802.16 based technology, also known as Worldwide Interoperability for Microwave Access (WiMAX), is one of the most promising and attractive alternatives for last mile broadband wireless access. As expected, IEEE 802.16 standard and its evolutions have been developed to deliver a variety of multimedia applications with different Quality-of-Service (QoS) requirements, such as throughput, delay, delay jitter, fairness and packet loss rate. The physical layer specifications and MAC signaling protocols have been well defined in the standard (2), however, radio resource management (RRM), i.e., scheduling and call admission control, still remains as an open issue, which plays an important role in QoS provisioning for different types of service.

Orthogonal Frequency Division Multiple Access (OFDMA) is a physical layer specification for IEEE 802.16 systems. OFDMA builds on Orthogonal Frequency Division Multiplexing (OFDM), which is immune to intersymbol interference and frequency selective fading, as it divides the frequency band into a group of mutually orthogonal subcarriers, each having a much lower bandwidth than the coherence bandwidth of the channel. In multi-user environment, OFDMA provides another degree of freedom by enabling dynamic assignment of subcarriers to different users at different time instances, to take advantage of the fact that at any time instance channel responses are different for different users at different subcarriers (3). Thus, dynamic subcarrier assignment (DSA) and adaptive power allocation (APA) to multiple users can be employed to improve the system performance significantly.

Recently, radio resource management for OFDMA systems has attracted enormous research interests. Many scheduling algorithms have been proposed which can adapt to changes in users' channel conditions and QoS requirements. In the literature, the resource allocation problem can be divided into two categories with different objectives. The objective of the first category is to minimize the total transmit power subject to individual data rate constraints, see (7)-(9). The objective of the second category aims at maximizing the overall (weighted) transmission rate subject to power constraints, see (10)-(12). In either case, the optimal resource allocation solutions are difficult to get due to high computational complexity. Instead, suboptimal solutions based on relaxation, problem splitting, or heuristic algorithms are proposed to reduce computational complexity (4). Such algorithms are often refereed to as *loading algorithms*.

In most loading algorithms, the QoS requirement of each user is usually defined in terms of a fixed data rate per frame. However, in practical communication systems, it is neither sufficient nor efficient to represent different QoS requirements by a fixed data rate per frame. The resource allocation problem for systems supporting both real-time (RT) and non-real-time (NRT) multimedia traffic becomes much more complicated when diverse QoS requirements have to be considered. The transmission of RT packets can be delayed as long as the delay constraint is not violated, and the transmission of NRT packets can be more elastic. Furthermore, most loading algorithms assume that users always have data to transmit, which is not the case in real systems. Instead, appropriate traffic models should be taken into account in the design of scheduling algorithms. Therefore, efficient *packet-based scheduling algorithms* are of interest. Many packet scheduling algorithms with different design objectives have been proposed in (13)-(22).

With respect to packet-based scheduling algorithms, most of the existing literature focuses on the design of one-level flat scheduler. In such approach, each connection is assigned a priority value based on some criterion and the connection with the highest priority value is scheduled for transmission. This approach has the advantage of low implementation complexity. However, due to different traffic patterns and diverse QoS requirements among different service classes, it is hard to well define a unified priority criterion that can work well for all service classes. Thus, it is desirable to individually design the scheduling algorithm for each service class and separate the resource allocation from the packet scheduling. The first paper proposing the idea of a two-level hierarchical scheduler is in (23). Performance comparisons between one-level and two-level schedulers are done in (24). In (25), Chang et al. proposed an adaptive hierarchical polling approach to minimize the average polling delay and bandwidth used for polling. However, so far little work has been done in the design of an efficient bandwidth distribution algorithm for the aggregate resource allocator, which is critical on the overall performance of a two-level hierarchical scheduler and therefore should be carefully designed.

In this chapter, we present a downlink resource management framework for QoS scheduling in OFDMA based WiMAX Systems. The framework consists of a dynamic resource allocation (DRA) module and a connection admission control (CAC) module. DRA emphasizes on how to share the limited radio resource in terms of subchannels and time slots among subscriber stations with the objective of increasing the spectral efficiency while satisfying the diverse QoS requirements in each service class. CAC highlights how to limit the number of ongoing connections preventing the system capacity from being overused. The major contributions of this chapter include:

1. A two-level hierarchical scheduler is employed to split the resource allocation problem into two subproblems: a bandwidth distribution problem in the aggregate resource allocator and a scheduling problem in class schedulers. Yet there is sufficient coupling between the allocator and the schedulers as the allocator is aware of the performance of the schedulers.
2. A novel priority-based scheduling algorithm is proposed for rtPS and nrtPS class schedulers, which tightly couples the packet scheduling and subcarrier allocation together through in integrated cross-layer approach to take advantage of the inter-dependencies between the PHY and MAC layers.
3. An adaptive estimation-based bandwidth distribution scheme is proposed for the aggregate resource allocator. The proposed scheme first estimates the required amount of bandwidth in each class scheduler based on the backlogged traffic and the average modulation efficiency. Then an exponentially smoothed curve with respect to QoS satisfaction is applied to adjust the estimation in order to increase the spectral efficiency while maintaining a guaranteed QoS performance.
4. An effective measurement-based connection admission control policy is proposed for the CAC module, which takes the current state of the network and class priority into consideration when admission decisions are made.

Through the detailed system-level simulations, the study shows that the proposed resource management framework can significantly increase the spectral efficiency while ensuring the QoS requirements of each service class.

The rest of the chapter is organized as follows: We first give a brief introduction of IEEE 802.16 in Section 2. Then, the structure of the proposed downlink resource management framework is described in Section 3, followed by the design of hierarchical resource allocation algorithms and connection admission control policies in Section 4 and Section 5, respectively. In Section 6, simulation environments and results are outlined and discussed. Finally, conclusions are drawn in Section 7.

2. Overview of IEEE 802.16 Networks

IEEE 802.16/WiMAX technology supports both mesh and point-to-multipoint (PMP) networks (2). In the PMP mode, the network has a cellular structure where base station (BS) governs all the communications in the network and the subscriber stations (SSs) cannot communicate with each other directly. In contrast, in the mesh mode, traffics can be exchanged

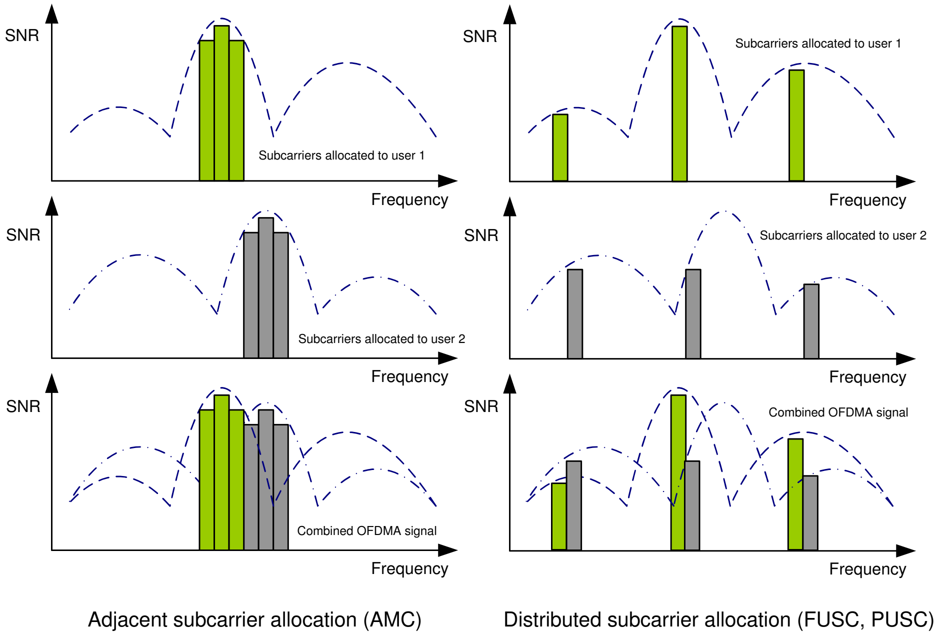
SNR
Subcarriers allocated to user 1
Frequency
SNR
Subcarriers allocated to user 2
Frequency
SNR
Combined OFDMA signal
Frequency
SNR
Subcarriers allocated to user 1
Frequency
SNR
Subcarriers allocated to user 2
Frequency
SNR
Combined OFDMA signal
Frequency
Adjacent subcarrier allocation (AMC)
Distributed subcarrier allocation (FUSC, PUSC)

Fig. 1. Adjacent and distributed subcarrier allocation

directly between SSs. In this chapter, we only concentrate on the WiMAX PMP network. The physical layer of the IEEE 802.16 air interface operates either at 10-66 GHz for line-of-sight (LOS) communications or 2-11 GHz for non-line-of-sight (NLOS) communications, and it supports data rates in the range of 32-130 Mbps depending on the transmission bandwidth as well as the modulation and coding schemes used (5). Adaptive modulation and coding scheme (AMC) is supported in the standard. The design objective of AMC is to maximize the data rates by adjusting the transmission parameters according to time-varying channel conditions, while maintaining a prescribed target packet error rate (PER).

As specified in the standard, IEEE 802.16 employs OFDM in the physical layer. In particular, two different air interfaces based on OFDM can be used: WirelessMAN-OFDM and WirelessMAN-OFDMA. The first option employs fast Fourier transform (FFT) of size 256 (subcarriers). Time-division multiplexing (TDM) and time-division multiple access (TDMA) are used for downlink and uplink transmission respectively. The second option employs a larger FFT space (2048 and 4096 subcarriers) which are further grouped into subchannels. The subchannels are assigned to different subscriber stations and may employ different modulation and coding schemes to exploit frequency diversity as well as time diversity. The subchannels are also used for multiple access, namely, orthogonal frequency division multiple access (OFDMA). There are two approaches of allocating subcarriers to form a subchannel: distributed subcarrier permutation and adjacent subcarrier permutation. The two approaches are shown in Fig. 1. In distributed subcarrier permutation, a subchannel is formed with different subcarriers randomly distributed across the channel spectrum. This approach maximizes the frequency diversity and averages inter-cell interference. It is suitable for mobile environment where channel characteristics change fast. Both partial usage of subchannels (PUSC) and full usage of subchannels (FUSC) schemes employ distributed subcarrier permutation. In adjacent subcarrier permutation, a subchannel is formed by grouping adjacent subcarriers. This approach creates a 'loading gain' and is easy to use with beam-forming adaptive antenna system (AAS). It is suitable for stationary or nomadic environment where channel characteristics change slowly. The AMC scheme employs adjacent subcarrier permutation.

IEEE 802.16 standard supports both frequency-division duplex (FDD) and time-division duplex (TDD) transmission modes. For FDD scheme, distinct frequency channels are assigned for uplink and downlink transmissions. In contrast, TDD scheme uses a single frequency channel for uplink and downlink transmissions by dividing the MAC frame into uplink and downlink subframes. The length of these subframes are determined dynamically by the BS and are broadcasted to the SSs through downlink and uplink MAP messages (DL-MAP and UL-MAP) at the beginning of each frame.

Four types of services are defined in the standard, each of which has different QoS requirements (2):

- Unsolicited grant service (UGS): This type of service is designed to support real-time service flows, with strict delay requirement, which generate fixed-size data packets periodically, such as T1/E1.
- Real-time polling service (rtPS): This type of service is designed to support real-time service flows, with less stringent delay requirements, which generate variable-size data packets at periodic intervals, such as VoIP with silence suppression.
- Non-real-time polling service (nrtPS): This type of service is designed to support delay-tolerant data streams which are more bursty in nature, such as FTP. In general, the nrtPS can tolerate longer delays and is insensitive to delay jitter, but requires a minimum throughput.
- Best-effort service (BE): This type of service is designed for traffic with no QoS requirements, such as email, and therefore may be handled on a resource-available basis.

3. Downlink Resource Management Framework

3.1 Assumptions and Preliminaries

In this chapter, we only investigate the WiMAX PMP network with OFDMA-TDD operation. We assume that subscriber stations are stationary or nomadic users with slowly varying channel conditions. Adjacent subcarrier permutation strategy is employed to support AMC. In OFDMA, radio resource is partitioned in both frequency domain and time domain, which results in a hybrid frequency-time domain resource allocation. It provides an added dimension of flexibility in terms of higher granularity compared to OFDM/TDM systems.

We consider the downlink scenario of an infrastructure-based OFDMA system with U_s subcarriers and K users. At the physical layer, the time axis is divided into frames with fixed length, each of which consists of a downlink (DL) and an uplink (UL) subframe to support TDD operation. In each DL subframe, there are U_t time slots available for downlink transmissions, each of which may contain one or several OFDM symbols. To reduce the resource addressing space, channel coherence in frequency and time is exploited by grouping I_s adjacent subcarriers and I_t time slots to form a basic resource unit (BRU) for resource allocation. A BRU is the minimum resource allocation unit as shown in Fig. 2. The size of a BRU is adjusted so that the channel experiences flat fading in both frequency and time domain. Thus in each

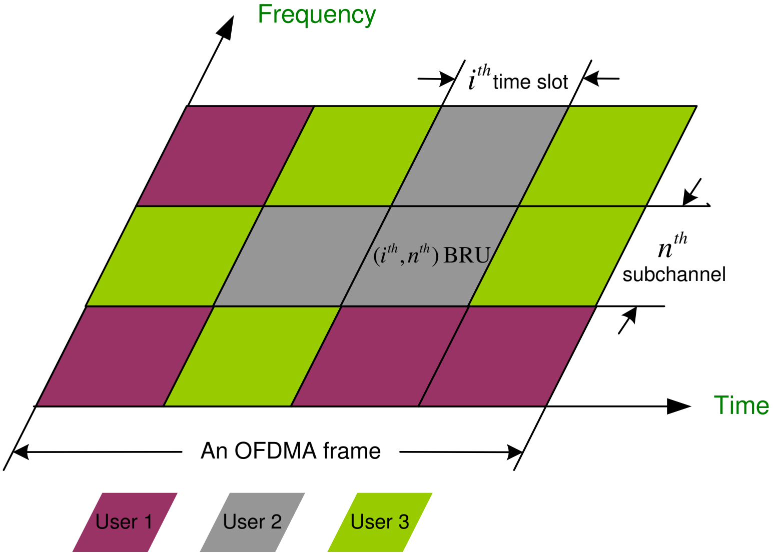

Fig. 2. Frequency-time domain radio resource allocation in OFDMA systems

DL subframe, there are $S = U_s/I_s$ subchannels in frequency domain and $N = U_t/I_t$ slots in time domain, which corresponds to a total of $S * N$ BRUs available in frequency-time domain for DL transmissions. Each BRU can be assigned to different users and be independently bit and power loaded. In principle, adaptive power allocation in each BRU can improve the system performance. However, some studies show that performance improvements are only marginal over a wide range of SNRs due to the statistical effects (3). Therefore, we assume that the total transmission power is equally distributed among all subchannels.

We further assume that in each frame the base station has perfect knowledge of channel state information (CSI) for each subchannel of each user. This can be obtained by piggybacking such information in each uplink packet, which is suitable for slowly varying channels. Based on CSI, adaptive modulation and coding scheme is employed to adjust the transmission mode dynamically according to the time-varying channel conditions. Multiple transmission modes are available, with each mode representing a pair of specific modulation format and a forward error correcting code. The transmission mode is determined by the instantaneous signal-to-noise ratio (SNR). To utilize the PHY layer resources more efficiently, fragmentation at the MAC layer is enabled. A separate queue with a finite queue length of L MAC protocol data units (PDUs) is maintained for each connection at the base station. We assume that the MAC PDUs are of fixed size, each of which contains d information bits.

3.2 Structure of the Resource Management Framework

The proposed downlink resource management framework consists of a dynamic resource allocation (DRA) module and a connection admission control (CAC) module. CAC is responsible for preventing the system capacity from being overused by limiting the number of ongoing connections. DRA aims at an efficient usage of the scarce radio resource, while maintaining

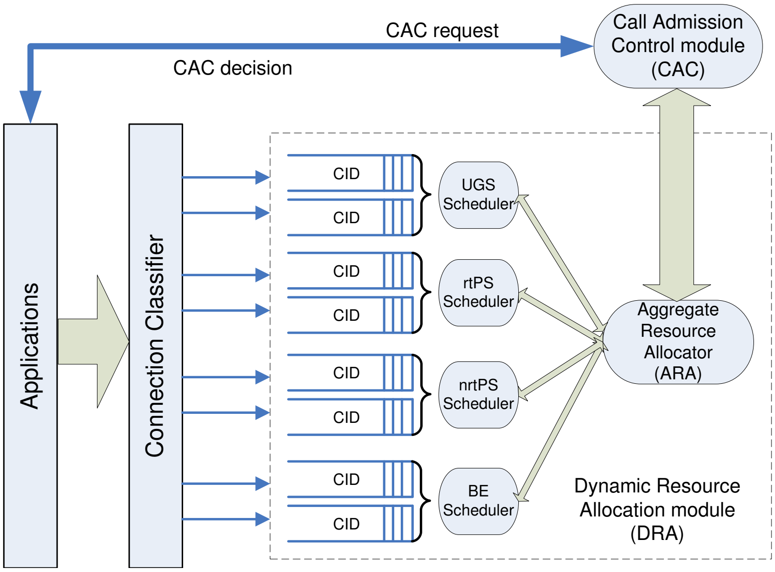

Fig. 3. Structure of the proposed downlink resource management framework for IEEE 802.16 systems

a guaranteed QoS performance among all admitted users. From a cross-layer perspective, resource management should be a joint optimization of a large number of variables ranging from application layer to physical layer. That is, which user should be scheduled for transmission in which BRU, under certain QoS constraints and time-varying channel conditions. However, this would lead to a very complex algorithm design since four types of services with different QoS requirements are defined in the standard. This approach would also lose flexibility if new traffic requirements or different optimization goals were to be considered.

Therefore, we adopt a two-level hierarchical scheduler for the DRA module, a loosely cross-layer approach trying to strike a balance between flexibility and modularity. Then the resource allocation problem can be split into two subproblems, i.e., a bandwidth distribution problem in the aggregate resource allocator and a scheduling problem in class schedulers. Yet there is sufficient coupling between the allocator and the schedulers as the allocator is aware of the performance of the schedulers. An advantage of this two-level hierarchical resource allocation architecture is that the algorithms for the allocator and the schedulers can be developed independently of each other. As an example, if the scheduling algorithm of a class scheduler is changed from, let's say, maximum SNR to proportional fairness, this will affect the way of BRU assignment in that class scheduler, but the bandwidth distribution algorithm in the aggregate resource allocator can be kept the same, as long as the design objective remains unchanged.

Fig. 3 depicts the proposed downlink resource management framework for IEEE 802.16 systems. When an application initiates a connection, it sends the connection request to the CAC module with connection type, traffic parameters, and QoS requirements. Then the CAC module interacts with the DRA module to get the current network state and commits admission decisions. All arriving packets from the application layer are classified by the connection classifier according to their connection identifications (CID) and traffic types, and are sent to the corresponding service class and get queued. The DRA module is responsible for scheduling packets of all admitted connections. It consists of an aggregate resource allocator (ARA) and four class schedulers. The ARA distributes bandwidth to each class scheduler based on some criterion. Once the class scheduler receives bandwidth from the ARA, it schedules packets in its queues. In each class scheduler, because the incoming flows have similar traffic patterns and QoS requirements, the class scheduler has the freedom to independently choose its own scheduling algorithm which can best meet the QoS requirements. Therefore, this two-level hierarchical resource allocation module can have multiple scheduling criteria and better schedule packets in each service class than its one-level flat counterpart.

4. Hierarchical Resource Allocation

In this section, we first describe the scheduling algorithms employed in each class scheduler, then an adaptive estimation-based bandwidth distribution algorithm for the aggregate resource allocator is proposed.

4.1 Scheduling Algorithms for Class Schedulers

Class scheduler in each service class receives bandwidth from the ARA and involves in the allocation of subchannels and time slots among different users in its service queues. Scheduling algorithms designed for class schedulers should have the goal of maximizing the spectral efficiency with satisfied QoS performance. In this section, we apply the appropriate packet scheduling algorithm to each class scheduler.

4.1.1 Scheduling UGS connections

The scheduling of UGS connections is well defined by the standard. In UGS, the transmission mode at the PHY layer is fixed during the whole service time (2). The AMC is not employed for UGS connections. The time slots allocated for UGS connections per frame are fixed, based on their constant bit-rate requirements negotiated in the initial service access phase.

4.1.2 Scheduling rtPS and nrtPS connections

The rtPS connection is delay-sensitive and has strict delay requirement. The nrtPS connection can tolerate longer delays, but requires a minimum throughput. We propose a novel priority-based scheduling algorithm for rtPS and nrtPS class schedulers. The basic idea behind the proposed algorithm is that the transmission is scheduled in a packet-by-packet fashion based on its priority value. Specifically, at each scheduling interval, if a PDU was scheduled for transmission on a specific subchannel, it is assigned a priority value based on the instantaneous channel condition (PHY layer issue), as well as the QoS constraint (MAC layer issue). Then we can formulate the scheduling problem into a mathematical optimization problem with the objective to maximize the total achievable priority values.

We apply an extended EXP algorithm as our priority function for both rtPS and nrtPS connections. The EXP rule was proposed to provide QoS guarantees over a shared wireless link in terms of the average packet delay for RT traffic and a minimum throughput for NRT traffic (19).

For rtPS connections, if the i^{th} PDU from the k^{th} connection is scheduled for transmission on subchannel n, its priority is calculated as:

$$\mathbf{P}(k,i,n) = a_k \cdot \frac{\mu_{k,n}(t)}{\overline{\mu}_k(t)} \cdot \exp\left(\frac{a_k W_{k,i}(t) - \overline{aW}}{1 + \sqrt{\overline{aW}}}\right) \tag{1}$$

where $\overline{aW} = \frac{1}{K}\sum_k a_k W_{k,1}(t)$, and $a_k = -\log \delta_k / T_{k,\max}$. $W_{k,i}(t)$ is the i^{th} PDU delay of connection k at time t, $T_{k,\max}$ is the maximum allowable delay of connection k, δ_k is the maximum outage probability of connection k, $\mu_{k,n}(t)$ is the instantaneous channel rate with respect to the signal-to-noise ratio (SNR) and a predetermined target error probability if subchannel n is assigned to connection k at time t, and $\overline{\mu}_k(t)$ is the exponential moving average (EMA) channel rate of connection k with a smoothing factor t_c, calculated as:

$$\overline{\mu}_k(t) = (1 - \frac{1}{t_c})\overline{\mu}_k(t-1) + \frac{1}{t_c}\mu_k(t) \tag{2}$$

where $\mu_k(t) = \sum_{n=1}^{N} c_{k,n} \cdot \mu_{k,n}(t)$ is the total channel rate of connection k at time t. If subchannel n is assigned to connection k, $c_{k,n} = 1$, otherwise $c_{k,n} = 0$.

For nrtPS connections, the extended EXP algorithm is used in conjunction with a token bucket control to guarantee a minimum throughput (19). We associate each nrtPS queue with a virtual token bucket. Tokens in each bucket arrive at a constant rate $r_{k,\text{req}}$, which is the required minimum throughput of connection k. After a PDU is scheduled for transmission, the number of tokens in the corresponding token queue is reduced by the actual amount of data transmitted. The calculation of the priority for an nrtPS PDU is similar to Exp. (1), with the exception that $W_{k,i}(t)$ in nrtPS is defined as the virtual waiting time of the i^{th} PDU from connection k:

$$W_{k,i}(t) = \frac{\max\left\{0, Q_k(t) - (i-1)\cdot d\right\}}{r_{k,\text{req}}} \quad k \in \text{nrtPS} \tag{3}$$

where $Q_k(t)$ is the number of tokens associated with connection k at time t, and d is the fixed size of a MAC PDU.

Let $\mathbf{u}(k,i,n)$ be defined as a binary random variable indicating subchannel allocation. That is, $\mathbf{u}(k,i,n) = 1$ means that the i^{th} PDU from connection k is allocated for transmission on subchannel n, and $\mathbf{u}(k,i,n) = 0$ otherwise. Also let us define $\mathbf{m}(k,i,n)$ be the number of time slots occupied on subchannel n if the i^{th} PDU from connection k is scheduled for transmission on subchannel n, calculated as:

$$\mathbf{m}(k,i,n) = \lceil \frac{d}{\mu_{k,n}(t)} \rceil \tag{4}$$

where $\lceil x \rceil$ denotes the smallest integer larger than x.

Then, the scheduling problem can be mathematically formulated as follows:

$$\arg\max_{\mathbf{u}(k,i,n)} \sum_{k=1}^{K}\sum_{i=1}^{L}\sum_{n=1}^{S} \mathbf{u}(k,i,n) \cdot \mathbf{P}(k,i,n) \tag{5}$$

subject to:

$$\sum_{k=1}^{K}\sum_{i=1}^{L} \mathbf{u}(k,i,n) \cdot \mathbf{m}(k,i,n) \leq N \quad \forall n \tag{6}$$

$$\sum_{n=1}^{S} \mathbf{u}(k,i,n) \leq 1 \quad \forall k,i \tag{7}$$

$$\mathbf{u}(k,i,n) \in \{0,1\} \quad \forall k,i,n \tag{8}$$

where S denotes the total number of subchannels, N denotes the total number of time slots, K denotes the total number of connections, and L denotes the maximum queue size.

The first constraint ensures that the allocated bandwidth does not exceed the total available bandwidth in terms of time slots on each subchannel. The second constraint says that a PDU can only be transmitted via one subchannel. The instantaneous channel conditions and the QoS related parameters are embodied into the priority function $\mathbf{P}(k,i,n)$ with the objective of maximizing the total achievable priority values, thus improving the spectral efficiency while maintaining QoS guarantees.

The above optimization problem can be solved by determining the values of binary variable $\mathbf{u}(k,i,n)$ through standard linear integer programming (LIP)[1]. The solution to the problem provides an optimal resource allocation. However, the computation complexity of the optimal solution is too high to be applied in practical systems. To reduce the computational complexity, we propose a suboptimal algorithm with low complexity.

In the suboptimal algorithm, we allocate radio resources on a packet-by-packet basis. The general idea is that, at each scheduling interval, the packet with the highest priority value from all queues is scheduled for transmission, and this procedure continues until either there is no radio resource left or there is no packet remaining unscheduled in the queue. A detailed description of the proposed scheduling algorithm is listed in pseudocode 1, where Ω_s^k is the set of subchannels that are available for data transmission of connection k, t_n is the number of residual time slots on subchannel n, q_k is the current queue size of connection k, and i_k is a pointer to the next PDU to be scheduled of connection k.

[1] The optimal solution of the LIP problem formulated in this chapter is obtained by using the General Algebraic Modeling System (GAMS).

Algorithm 1 Suboptimal Scheduling Algorithm for rtPS and nrtPS Class Schedulers

```
1: Set $t_n \leftarrow N$ for $\forall n$ {initialize $t_n$}
2: Set $i_k \leftarrow 1$ for $\forall k$ {initialize $i_k$}
3: Get $q_k$ for $\forall k$ {get the queue size of connection $k$}
4: for $k = 1$ to $K$ do
5:    if $q_k > 0$ then
6:       Set $\Omega_s^k \leftarrow \{1, \cdots, S\}$ {initialize $\Omega_s^k$}
7:    else
8:       Set $\Omega_s^k \leftarrow \phi$ {set $\Omega_s^k$ to be null}
9:    end if
10: end for
11: while $\exists x, \Omega_s^x \neq \phi$ do
12:    for $k = 1$ to $K$ do
13:       while $\Omega_s^k \neq \phi$ do
14:          Select $n \leftarrow \arg\max_{n \in \Omega_s^k} \mu_{k,n}(t)$ {assign the best subchannel from the available subchannel set}
15:          if $t_n \geq \lceil \frac{d}{\mu_{k,n}(t)} \rceil$ then
16:             Calculate $\mathbf{P}(k, i_k, n)$ in Exp. (1)
17:             BREAK
18:          else
19:             $\Omega_s^k \leftarrow \Omega_s^k - \{n\}$ {remove $n$ from the available subchannel set if there is not enough capacity left}
20:             CONTINUE
21:          end if
22:       end while
23:    end for
24:    Schedule the $i_{k^*}$th PDU of connection $k^*$ on subchannel $n^*$, where $(k^*, i_{k^*}, n^*) \leftarrow \arg\max \mathbf{P}(k, i_k, n)$
25:    $t_{n^*} \leftarrow t_{n^*} - \lceil \frac{d}{\mu_{k^*,n^*}(t)} \rceil$ {update the residual time slots}
26:    if $i_{k^*} = q_k$ then
27:       $\Omega_s^{k^*} \leftarrow \phi$ {set $\Omega_s^{k^*}$ to be null when all pending PDUs of connection $k^*$ have been scheduled for transmission}
28:    else
29:       $i_{k^*} \leftarrow i_{k^*} + 1$ {point to the next pending PDU}
30:    end if
31: end while
```

It works as follows: If connection k has pending traffic in the queue, the proposed algorithm first pre-allocates the best subchannel n in terms of the instantaneous channel quality to connection k from its available subchannel set Ω_s^k (see Step 14). If there is not enough capacity left on the best subchannel n to accommodate one PDU from connection k's queue, subchannel n will be removed from connection k's available subchannel set, and the second best subchannel n' will be selected. This procedure continues until a best possible subchannel is pre-allocated to connection k (see Step 13-22). Otherwise, connection k is removed from the scheduling list. After the subchannel pre-allocation process for all connections is complete, the algorithm calculates the priority value of the head-of-line (HOL) PDU in each non-empty queue, and schedule the PDU with the highest priority value for transmission on subchannel n^* (see Step 16 & 24). The scheduled PDU is removed from the corresponding queue and the consumed radio resources in terms of time slots are subtracted on subchannel n^* (see Step 25 & 26-30). Then it starts from the beginning and continues until either there is no radio resource left or there is no PDU pending in the queue. A detailed description of the proposed suboptimal algorithm is listed in pseudocode 1.

4.1.3 Scheduling BE connections

Since there are no QoS guarantees for BE connections, we simply apply the Proportional Fair (PF) algorithm to schedule BE traffic. The PF algorithm attempts to serve each user at its peak channel condition. Hence the PF algorithm can utilize the radio resource efficiently and give proportional fairness among users (20). At each scheduling point, the PF algorithm selects connection k for transmission on subchannel n as follows:

$$(k,n) = \arg\max_{k} \frac{\mu_{k,n}(t)}{\overline{\mu}_k(t)} \quad \forall n \tag{9}$$

where $\mu_{k,n}(t)$ and $\overline{\mu}_k(t)$ are defined as the same in (1).

4.2 Scheduling Algorithm for the Aggregate Resource Allocator

The aggregate resource allocator (ARA) is responsible for distributing the total available bandwidth among class schedulers. If the ARA does not allocate enough bandwidth to the class scheduler, the QoS requirements in the corresponding service class may not be guaranteed. On the other hand, if the ARA allocates too much bandwidth to the class scheduler, the allocated radio resource may not be utilized efficiently or even be wasted. Therefore, the resource distribution algorithm of ARA is a critical factor on the performance of class schedulers and has to be carefully designed.

4.2.1 Conventional Bandwidth Distribution Algorithms

One possible solution is that the ARA distributes bandwidth among service classes following strict class priority, from highest to lowest, i.e., UGS, rtPS, nrtPS and BE. After all connections in high priority class have been served, connections in low priority class are scheduled for transmission. By doing so, the ARA can differentiate different service classes based on their class priority. The priority-based scheme is simple, but one disadvantage of this algorithm is that higher priority classes may starve the bandwidth for lower priority classes.

To overcome this problem, the ARA may partition the total bandwidth into several portions to satisfy proportional fairness among service classes. This method can prevent the starvation of low priority classes. There are static and dynamic bandwidth distribution schemes in this method. In the static scheme, the ARA distributes a fixed amount of bandwidth to each class scheduler at every scheduling interval. This approach has the advantage of simplicity and works well when the traffic pattern in each service class is regular and stable, which unfortunately is not always the case in data communications. Therefore, a dynamic bandwidth distribution scheme which can adapt to the traffic pattern and the performance of class schedulers is believed to be a better solution.

4.2.2 Proposed Adaptive Bandwidth Distribution Algorithm

The design objective of our proposed resource allocation algorithm is to adaptively allocate bandwidth to each service class in order to increase the spectral efficiency while satisfying the diverse QoS requirements. In designing the proposed adaptive resource allocation algorithm, we have taken the following aspects into account: (i) the backlogged traffic; (ii) the average modulation efficiency; (iii) the QoS satisfaction. The general idea is that, in the scheduling interval, the ARA first estimates the amount of bandwidth required in each class scheduler based on the backlogged traffic and the average modulation efficiency. Then depending on the QoS performance in each class scheduler, the estimation is further increased or decreased to maintain the guaranteed performance.

We separate the bandwidth allocation of UGS class from the others as it has been defined by the standard. At the beginning of each frame, the ARA allocates a fixed amount of time slots $N_{\text{UGS}} = \sum_{i\in\{\text{UGS}\}} \theta_i$ to UGS connections based on their constant bit-rate requirements negotiated in the initial service access phase, where θ_i is the number of time slots required by UGS connection i. Let N_{total} be the total number of time slots in each frame, then the residual time slots after serving UGS class $N_{\text{rest}} = N_{\text{total}} - N_{\text{UGS}}$ are distributed among rtPS, nrtPS and BE classes, which employ AMC scheme at the PHY layer.

For rtPS class, since each packet has a rigid delay requirement, the total sum of the current queue size in rtPS class is an appropriate measure for the backlogged traffic $B_{\text{rtPS}}(t) = \sum_{i\in\{\text{rtPS}\}} q_i(t)$, where $q_i(t)$ is the number of bits in queue i at time t. The average modulation efficiency $\overline{\mu}_{\text{rtPS}}(t)$ is defined as the average number of bits carried per OFDM symbol over a sliding time window t_c. γ is a QoS related parameter (i.e., maximum allowable delay in rtPS) representing the proportion of backlogged traffic that has to be transmitted within each frame. Then the estimated number of time slots required for rtPS class can be expressed as follows:

$$E_{\text{rtPS}}(t) = \alpha(t) \cdot \frac{\gamma B_{\text{rtPS}}(t)}{\overline{\mu}_{\text{rtPS}}(t)} \tag{10}$$

where $\alpha(t)$ is a QoS-aware adjustment factor that is updated according to the performance of the class scheduler on a frame by frame basis. The basic idea is that when the class scheduler experiences good QoS satisfaction, the value of $\alpha(t)$ is decreased to save bandwidth for other classes. Otherwise, the value of $\alpha(t)$ is increased to guarantee the required QoS. Towards this end, an exponentially smoothed curve is applied to adjust the value of $\alpha(t)$. The adjustment, which is $|\Delta\alpha(t)| = |\alpha(t) - \alpha(t-1)|$, is minor if the QoS outage probability is around a predefined target threshold. Otherwise, $|\Delta\alpha(t)|$ is exponentially increased to either increase or decrease the allocated bandwidth to the class scheduler, as illustrated in Fig. 4. The calculation of $\Delta\alpha(t)$ is specified as follows:

$$\Delta\alpha(t) = \begin{cases} \xi_{\max} \cdot \frac{\exp(\beta\cdot d(t))-1}{\exp(\beta\cdot D_{\max})-1} & \text{if } \mathrm{P_r}(t) \geq T_h \\ -\xi_{\max} \cdot \frac{\exp(\beta\cdot d(t))-1}{\exp(\beta\cdot D_{\max})-1} & \text{if } \mathrm{P_r}(t) < T_h \end{cases} \tag{11}$$

where $d(t)$ is the truncated difference between the current outage probability and the outage probability threshold:

$$d(t) = \min\{|\mathrm{P_r}(t) - T_h|, D_{\max}\}$$

where $\mathrm{P_r}(t)$ is the delay outage probability at time t, T_{h} is the outage probability threshold, $D_{\max}$ is the truncated maximum value of $|d(t)|$, β is a shape factor which is used to tune the adaptation degree, and $\xi_{\max}$ is the maximum value of $|\Delta\alpha(t)|$. Term $(\exp(\beta \cdot d(t)) - 1)/(\exp(\beta \cdot D_{\max}) - 1)$ is a normalization function of $(\mathrm{P_r}(t) - T_{\mathrm{h}})$. When $\mathrm{P_r}(t)$ is close to T_{h}, the normalized value is close to zero. Otherwise it increases exponentially to one. The overall bandwidth estimation procedure for rtPS class can be described as follows:

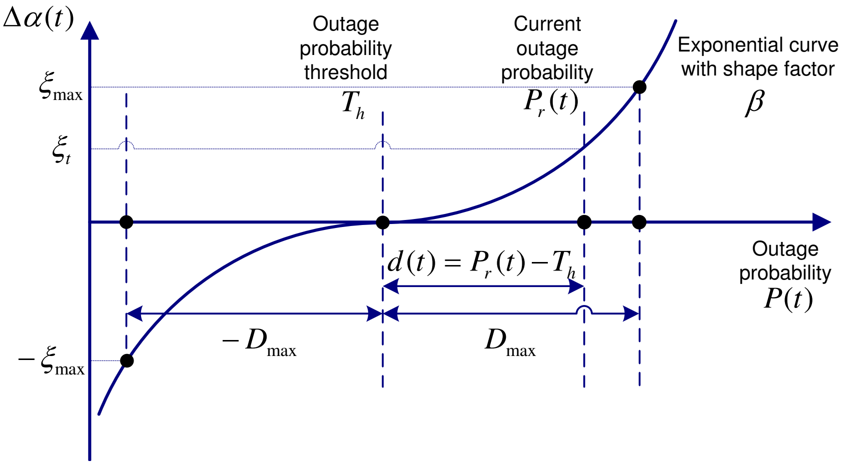

Fig. 4. An exponentially smoothed curve with respect to QoS satisfaction is applied to adjust the estimated amount of bandwidth in the class scheduler

- **Step 1:** At each scheduling instant, calculate the backlogged traffic $B_{\text{rtPS}}(t)$, the average modulation efficiency $\overline{\mu}_{\text{rtPS}}(t)$, and the current delay outage probability $\text{P}_{\text{r}}(t)$. Update the value of $\alpha(t)$:

$$\alpha(t) = \begin{cases} \min\{\alpha(t-1) + \Delta\alpha(t), \alpha_{\max}\} & \text{if } \text{P}_{\text{r}}(t) \geq T_{\text{h}} \\ \max\{\alpha(t-1) + \Delta\alpha(t), \alpha_{\min}\} & \text{if } \text{P}_{\text{r}}(t) < T_{\text{h}} \end{cases} \tag{12}$$

  where $\alpha_{\max}$ and $\alpha_{\min}$ are the maximum and minimum values of $\alpha(t)$, respectively.
- **Step 2:** Calculate the estimated bandwidth for rtPS class according to Exp. (10).

For nrtPS class, the bandwidth estimation procedure is the same as in rtPS class, except the definition of the backlogged traffic and the outage probability. Here we take the total number of virtual tokens associated with each queue as a measure for the backlogged traffic $B_{\text{nrtPS}} = \sum_{i \in \{\text{nrtPS}\}} v_i(t)$, where $v_i(t)$ is the number of virtual tokens in bucket i at time t. $\text{P}_{\text{r}}(t)$ in nrtPS is the throughput outage probability.

The proposed adaptive bandwidth distribution algorithm works as follows: At each scheduling instant, the ARA first allocates the amount of required bandwidth to UGS class (N_{UGS}) and a minimum amount of bandwidth to BE class ($N_{\text{BE}}^{\min}$). Once the ARA has estimated the amount of required bandwidth for rtPS and nrtPS classes (i.e., E_{rtPS} and E_{nrtPS}), it checks the remaining bandwidth. If the remaining bandwidth is larger than the estimated sum of E_{rtPS} and E_{nrtPS}, the ARA first allocates E_{rtPS} and E_{nrtPS} to rtPS and nrtPS class schedulers respectively. Then the residual bandwidth is distributed among rtPS, nrtPS and BE class schedulers proportional to their queue size Q_{rtPS}, Q_{nrtPS}, and Q_{BE}. Otherwise, if the remaining bandwidth is smaller than the estimated sum of E_{rtPS} and E_{nrtPS}, the ARA first allocates E_{rtPS} to rtPS class, then the residual bandwidth is allocated to nrtPS class. It is worth mentioning that in order to satisfy proportional fairness among class schedulers, each class scheduler is reserved a minimum amount of bandwidth. A detailed description of the proposed algorithm is listed in pseudocode 2.

Algorithm 2 Adaptive bandwidth distribution algorithm for the Aggregate Resource Allocator (ARA)

```
Set initial $N_{\text{total}}$ at the beginning of each frame
$N_{\text{UGS}} \leftarrow \sum_{i \in \{\text{UGS}\}} \theta_i$
Set $N_{\text{BE}}^{\text{min}}$
$N_{\text{residual}} \leftarrow N_{\text{total}} - N_{\text{UGS}} - N_{\text{BE}}^{\text{min}}$
Estimate the number of time slots required in rtPS class scheduler $E_{\text{rtPS}}$ by Exp.(10)
Estimate the number of time slots required in nrtPS class scheduler $E_{\text{nrtPS}}$ by Exp. (10)
if $N_{\text{residual}} \geq (E_{\text{rtPS}} + E_{\text{nrtPS}})$ then
    $N_{\text{residual}} \leftarrow N_{\text{residual}} - E_{\text{rtPS}} - E_{\text{nrtPS}}$
    $N_{\text{rtPS}} \leftarrow E_{\text{rtPS}} + N_{\text{residual}} \cdot \frac{Q_{\text{rtPS}}}{Q_{\text{rtPS}}+Q_{\text{nrtPS}}+Q_{\text{BE}}}$
    $N_{\text{nrtPS}} \leftarrow E_{\text{nrtPS}} + N_{\text{residual}} \cdot \frac{Q_{\text{nrtPS}}}{Q_{\text{rtPS}}+Q_{\text{nrtPS}}+Q_{\text{BE}}}$
    $N_{\text{BE}} \leftarrow N_{\text{BE}}^{\text{min}} + N_{\text{residual}} \cdot \frac{Q_{\text{BE}}}{Q_{\text{rtPS}}+Q_{\text{nrtPS}}+Q_{\text{BE}}}$
    if $N_{\text{rtPS}} < N_{\text{rtPS}}^{\text{min}}$ or $N_{\text{nrtPS}} < N_{\text{nrtPS}}^{\text{min}}$ then
        Adjust the values of $N_{\text{rtPS}}$, $N_{\text{nrtPS}}$ and $N_{\text{BE}}$ so that $N_{\text{rtPS}} \geq N_{\text{rtPS}}^{\text{min}}$ and $N_{\text{nrtPS}} \geq N_{\text{nrtPS}}^{\text{min}}$
    end if
else
    $N_{\text{rtPS}} \leftarrow \min\{E_{\text{rtPS}}, N_{\text{residual}}\}$
    $N_{\text{nrtPS}} \leftarrow N_{\text{residual}} - N_{\text{rtPS}}$
    $N_{\text{BE}} \leftarrow N_{\text{BE}}^{\text{min}}$
    if $N_{\text{rtPS}} < N_{\text{rtPS}}^{\text{min}}$ or $N_{\text{nrtPS}} < N_{\text{nrtPS}}^{\text{min}}$ then
        Adjust the values of $N_{\text{rtPS}}$ and $N_{\text{nrtPS}}$ so that $N_{\text{rtPS}} \geq N_{\text{rtPS}}^{\text{min}}$ and $N_{\text{nrtPS}} \geq N_{\text{nrtPS}}^{\text{min}}$
    end if
end if
```

5. Connection Admission Control

Connection admission control is a key component of QoS provisioning for wireless systems supporting multiple types of applications. It aims at maintaining the delivered QoS to different users at the target level by limiting the number of ongoing connections in the system. In this chapter, we propose a measurement-based approach as our CAC policy, of which a CAC decision is made depending on the current resource utilization of the network. When a new connection request is initiated, it informs the CAC module of the connection type (i.e., rtPS or nrtPS), the traffic parameters (i.e., arrival rate and service rate), and the QoS requirements (i.e., maximum delay or minimum throughput). Then the CAC module estimates the required amount of bandwidth ΔN to accommodate the incoming connection and performs a CAC decision based on the following conditions.

For UGS connections, as the transmission mode and the number of time slots allocated per connection per frame are negotiated in the initial service access phase and are fixed during the whole service time, a simple threshold-based CAC is applied:

$$N_{\text{UGS}}^{\text{current}} + \Delta N_{\text{UGS}} \leq N_{\text{UGS}}^{\text{max}} \tag{13}$$

where $N_{\text{UGS}}^{\text{current}}$ is the number of time slots occupied by the ongoing UGS connections, and $N_{\text{UGS}}^{\text{max}}$ is the maximum number of time slots that can be allocated to the UGS class scheduler. If this condition is satisfied, the incoming connection is accepted; otherwise, it is rejected.

For rtPS and nrtPS connections, when a new connection arrives, the CAC module interacts with ARA in the DRA module and gets the current estimated bandwidth occupied by the ongoing rtPS and nrtPS connections, $\overline{E}_{\text{rtPS}}(t)$ and $\overline{E}_{\text{nrtPS}}(t)$, which are the exponential moving average of $E_{\text{rtPS}}(t)$ and $E_{\text{nrtPS}}(t)$ mentioned in Eqn. (10). If the sum of the estimated bandwidth used by the ongoing rtPS and nrtPS connections ($\overline{E}_{\text{rtPS}}(t)$, $\overline{E}_{\text{nrtPS}}(t)$) and the estimated bandwidth to be used by the incoming connection (ΔN_{rtPS} or ΔN_{nrtPS}) is larger than a predefined upper threshold, the incoming connection is rejected; otherwise, the connection is accepted with certain probability depending on the estimated bandwidth usage and the connection priority. Specifically, when the estimated bandwidth occupancy is high or the priority of the incoming connection is low, the acceptance probability is small, and vice versa. A detailed description of the proposed CAC algorithm for rtPS connections is listed in pseudocode 3, where $N_{\text{th}}^{\text{max}}$ and $N_{\text{th}}^{\text{min}}$ are the maximum and minimum capacity threshold respectively, and $\rho_{\text{rtPS}} \in (0, 1]$ is a parameter that is used to differentiae class priorities. The same CAC algorithm is applied for nrtPS connections.

Algorithm 3 Connection admission control algorithm for rtPS connections

```
if $\overline{E}_{\text{rtPS}}(t) + \overline{E}_{\text{nrtPS}}(t) + \Delta N_{\text{rtPS}} > N_{\text{th}}^{\text{max}}$ then
    Reject the incoming connection
else if $\overline{E}_{\text{rtPS}}(t) + \overline{E}_{\text{nrtPS}}(t) + \Delta N_{\text{rtPS}} < N_{\text{th}}^{\text{min}}$ then
    Accept the incoming connection with probability $\rho_{\text{rtPS}}$
else
    Accept the incoming connection with probability
    $\rho_{\text{rtPS}} \cdot \frac{N_{\text{th}}^{\text{max}} - (\overline{E}_{\text{rtPS}}(t) + \overline{E}_{\text{nrtPS}}(t) + \Delta N_{\text{rtPS}})}{N_{\text{th}}^{\text{max}} - N_{\text{th}}^{\text{min}}}$
end if
```

For BE connections, they are always accepted since they do not impose any QoS constraints.

6. Simulation Results and Discussions

To evaluate the performance of the proposed downlink resource management framework for QoS scheduling in OFDMA based WiMAX networks, a system-level simulation is performed in OPNET.

6.1 System Model

We consider the downlink of a single-cell IEEE 802.16 system with OFDMA TDD operation. The cell radius is 2 km, where subscriber stations are randomly placed in the cell with uniform distribution. The total bandwidth is set to be 5 MHz, which is divided into 10 subchannels. The BS transmit power is set to 20W (43 dBm) which is evenly distributed among all subchannels. The duration of a frame is set to be 1 ms so that the channel quality of each connection remains almost constant within a frame, but may vary from frame to frame. The propagation model is derived from IEEE 802.16 SUI channel model (30). Path loss is modeled according to terrain Type A suburban macro-cell. Large-scale shadowing is modeled by log-normal distribution with zero mean and standard deviation of 8 dB. Small-scale shadowing is modeled by Rayleigh fading.

Table 1 summarizes the system parameters used in the simulation. We assume that all MAC PDUs are transmitted and received without errors and the transmission delay is negligible.

Parameters	**Value**
System	OFDMA/TDD
Central frequency	3500 MHz
Channel bandwidth	5 MHz
Number of subchannels	10
User distribution	Uniform
Beam pattern	Omni-directional
Cell radius	2 km
Frame duration	1 ms
BS transmit power	20 W
Thermal noise density	−174 dBm/Hz
Propagation model	802.16 SUI-5 Channel model
Maximum MAC PDU size	256 bytes

Table 1. A summary of system parameters

Modulation scheme	**Coding rate**	**bits/symbol**	**Target SNR for 1% PER (dB)**
BPSK	1/2	0.5	1.5
QPSK	1/2	1	6.4
QPSK	3/4	1.5	8.2
16QAM	1/2	2	13.4
16QAM	3/4	3	16.2
64QAM	1/2	4	21.7
64QAM	3/4	4.5	24.4

Table 2. Modulation and coding schemes for 802.16 (27)

The modulation order and coding rate in AMC is determined by the instantaneous SNR of each user on each subchannel. We follow the AMC table shown in Table 2, which specifies the minimum SNR required to meet a target packet error rate, e.g., 1%.

6.2 Traffic Model

In the simulation, different types of traffic sources are generated: VoIP, videoconference, and Internet traffic. VoIP and videoconference are served in UGS class and rtPS class, respectively. Internet traffic is served in nrtPS class and BE class. Each user alternates between the states of idle and busy, which are both exponentially distributed, and generates one or several traffic types independently during the busy period. VoIP traffic is modeled as a two-state Markov ON/OFF source (16). A videoconference consists of a VoIP source and a video source (16). Internet traffic can be web browsing that requires large bandwidth and generates bursty data of variable size. We apply the Web browsing model for the Internet traffic (17). A summary of traffic parameters for different traffic types are listed in Table 3.

6.3 Performance Evaluation

Since the performance of fixed bandwidth allocation for UGS connections is well defined by the standard and BE connections do not have any specific QoS requirements, here we only focus on the performance evaluation of rtPS and nrtPS connections. The delay constraint for