Designing and Implementing an IEEE 802.16 Network Simulator for Performance Evaluation of Bandwidth Allocation Algorithms

Yuan-Cheng Lai and Yen-Hung Chen
Department of Information Management
National Taiwan University of Science and Technology, Taiwan
layc@cs.ntust.edu.tw, D9609101@mail.ntust.edu.tw

Abstract

IEEE 802.16 technology is a promising technology to solve the last mile problem by its wireless, long-distance, and high-bandwidth transmission. However, there isn’t any suitable simulator to evaluate the performance between different IEEE 802.16 bandwidth allocation algorithms. In this paper, an IEEE 802.16 network simulator, Light WiMAX simulator (LWX), is proposed to provide flexible and robust functionalities for evaluating varies IEEE 802.16 bandwidth allocation algorithms. The contributions of LWX are 1) to support QoS, OFDMA, and multi-hop relay, 2) to provide a dynamic binding mechanism for the users to “plug-and-play” different algorithms without modifying and recompiling LWX, and 3) to supply the complete sample source codes and rich simulation scripts to let the users easily and quickly understand the architecture of LWX and integrate their works into LWX. Thus, it can benefit academic researchers and industrial developers to verify their designed algorithms and mechanisms.

Keywords

IEEE 802.16, IEEE 802.16j, Relay, Bandwidth Allocation, Path Selection, Simulator

1. Introduction

IEEE 802.16 is expected to be a revolution of broadband access technology to solve the network infrastructure design and expansion problems, which are how to construct a new network and how to expand an existing network to service the growing demand of the users [1]. The 802.16 network is generally constructed by a base station (BS) and some subscriber stations (SSs) with broadband wireless technology to connect each other. The BS provides connectivity, radio resource management, and control of SSs, and the SS supports the connectivity between subscribers and the BS [2]. However, the larger distance between the SS and BS is, the lower received signal strength and transmission rate the SS suffers. In order to improve the robustness of the network, a new station, called the relay station (RS), is introduced to extend the radio coverage and enhance the data transmission rate by exploiting the multi-hop relaying technique [3].

IEEE 802.16 has lots of distinguished features including QoS supports, OFDMA transmission, multi-hop relay, et al. To provide QoS supports for many network applications in IEEE 802.16 network, such as HDTV, video conference, and conventional Internet applications, the bandwidth is dispensed to each user by the bandwidth request-grant mechanism. That is, a SS acquires bandwidth from the BS by a bandwidth request (BW request) and the BS centrally decides the data transmission path and schedules the proper bandwidth to the SS according to its QoS requirements, channel quality, and the bandwidth requests of all connection. Furthermore, IEEE 802.16 adopts Orthogonal Frequency Division Multiple Access (OFDMA) transmission technique which groups all data subcarriers to form a number of subchannels. Thus, by adopting OFDMA, the SS can use some certain subchannels and other SSs can use other subchannels to transmit data simultaneously. This leads that the bandwidth resource can be well utilized via two-dimensional allocation of which dimensions are subchannel and time symbol. Therefore, the bandwidth allocation not only has to address the relay path selection and the bandwidth...
scheduling issues but also needs to consider the method to construct a two-dimensional bandwidth.

The algorithm with regard to the bandwidth allocation is not defined in the standard [2] [3]. Nevertheless, many algorithms were proposed including how to select the path [4], schedule the bandwidth [5], and construct the two-dimensional bandwidth [6] for each subscriber. It is impossible to evaluate these proposed algorithms in real IEEE 802.16 network environment because of the enormous cost of equipments. Thus, a well designed IEEE 802.16 network simulator is needed.

Although many simulators provide IEEE 802.16 simulation module, it is quite difficult to apply these modules directly to evaluate the performance of the proposed bandwidth allocation algorithms. For the IEEE 802.16 module provided by OPNET [7] or QualNet [8], the researchers would not known the detailed and actually supported functionalities until the expensive source code is purchased. For the most popular network simulator ns2 [9], many researchers designed and released their IEEE 802.16 module freely. However, the modules provided by NIST [10], NDSL [11], and IRC [12] do not support QoS, OFDMA, and multi-hop relay functionalities. Therefore, the existing IEEE 802.16 network simulators either have enormous cost and merely provide rough and sketchy specification or do not fully support IEEE 802.16 bandwidth allocation mechanism.

In this paper, we design and implement an IEEE 802.16 network simulator with a light-weight software architecture, named Light WiMAX simulator (LWX), which provides flexible and robust functionalities to evaluate the performance of IEEE 802.16 bandwidth allocation algorithms. The light-weight design of LWX is inherited from the original classes in ns2 to simplify simulator software architecture, accelerate the understanding of the operations and workflows of LWX, and cooperate with other network protocol components of ns2. LWX supports QoS, OFDMA, multi-hop relay specified in the standard [2] [3] to let the users test and verify their designs. Furthermore, the development of LWX is based on an object-oriented approach which allows the users to “plug-and-play” different algorithms without modifying the core of LWX. Furthermore, the users can just only change the settings of the simulation script without recompiling the simulator to observe what performance the IEEE 802.16 network systems can achieve by combining different relay path selection, bandwidth scheduling, and two-dimensional bandwidth constructing algorithms. In addition, to facilitate the users to use and integrate their algorithms into LWX, we also give complete sample source codes and rich simulation scripts of four algorithms including relay path selection, bandwidth scheduling, and two-dimensional bandwidth constructing algorithm.

The rest of this paper is organized in the following. In Section 2, some basic concepts of IEEE 802.16 are introduced. Section 3 formally describes the detailed design of LWX, and Section 4 evaluates LWX. Finally, the conclusion is given in Section 5.

2. IEEE 802.16 Network

The IEEE 802.16 protocol stack consists of two layers, physical layer and MAC layer. The former provides the means to transfer raw data, and the later is designed to support the physical layer to efficiently manage radio resource. With regard to the physical layer adopting OFDMA, three modulations, quadrature phase shift keying (QPSK), 16 quadrature amplitude modulation (16QAM), and 64 quadrature amplitude modulation (64QAM) with different coding rate are used according to the channel quality, i.e. carrier-to-interference-plus-noise ratio (CINR). Three duplex modes are used in OFDMA including TDD (Time Division Duplex), FDD (Frequency Division Duplex), and H-FDD (Half-duplex Frequency Division Duplex). TDD is the most attractive duplex mode because of its flexibility [13].

As shown in Figure 1 as an example, an IEEE 802.16 frame is a complete set for downlink and uplink transmissions and divided into DL and UL subframes in the time domain by using TDD. Each subframe contains an access zone and relay zone, where the former is for data transmission directly to or from SS and the latter is used to transmit data between BS and RS for relay usage. A burst is an allocated two-dimensional bandwidth assigned to one dedicated SS, and it is formed by slots. A slot is the minimal possible bandwidth allocation unit defined in the frequency and the time domain, and it contains one subchannel and one to three symbols where a symbol is the smallest allocation unit in the time domain and a subchannel is the smallest logical allocation unit in the frequency domain. With regard to the IEEE
802.16 frame structure, Preamble provides synchronization usage for each SS, FCH (frame control header) specifies the burst profile and the length of the burst that immediately follows the FCH, and the DL/UL/R-MAP describes each burst in time and frequency domain.

For the bandwidth request-grant mechanism in MAC layer, the BS needs to allocate bandwidth to each SS before SS receives or transmits data. On the downlink (from BS to SS), the BS simply allocates the bandwidth based on its policy and broadcasts the data packets to all SSs, and then each SS picks up the packets destined to it. On the uplink (from SS to BS), SS needs to send a bandwidth request (BW Request) to BS to acquire bandwidth for data transmission. Two ways to send a BW Request are contention-based and contention-free. In the contention-based mode, the SS sends a BW Request to BS for TXOP (transmission opportunity) based on TBEB (truncated binary exponential backoff algorithm) during contention periods. In the contention-free mode, the BS polls each SS and the SS sends its BW Request after receiving the polling signal. After receiving the BW Request, BS would dispense proper bandwidth to the SS.

In order to support QoS, the IEEE 802.16 MAC layer defines five service classes including Unsolicited Grant Service (UGS), Real-Time Polling Services (rtPS), Extended Real-Time Polling Services (etrPS), Non-Real-Time Polling Services (nrtPS), and Best Effort (BE). Each service class has its specific QoS parameters, including Maximum Sustained Traffic Rate ($R_{\text{max}}$), Minimum Reserved Traffic Rate ($R_{\text{min}}$), Maximum Latency ($L_{\text{max}}$), Tolerated Jitter, and Traffic Priority [2]. $R_{\text{max}}$ defines the peak rate, $R_{\text{min}}$ means the minimal sustainable rate, and $L_{\text{max}}$ specifies the maximum latency between the ingress time of a packet to the MAC layer and the forwarding time to its air interface.

### 3. The Design of LWX

Our developed IEEE 802.16 simulation module, called Light WiMAX (LWX) simulator, is accordance with the specifications of IEEE 802.16 [2] and IEEE 802.16j [3] standards and based on ns2 version 2.29 [9]. In this section, we first describe the software architecture and workflow of LWX, and then present the core design of LWX.

#### 3.1. The Software Architecture and Workflow

The software architecture of LWX and the relation between LWX and the original ns2 system are shown in Figure 2. Within LWX, all components including Traffic Handler, MAC Handler, PHY Handler, LWX Otc1 Script Transformer, and LWX Simulation Log Generator, are designed by object-oriented programming language C++ and grouped into several classes. Traffic Handler addresses the aggregation and mapping of the traffic from application layer of ns2 or from MAC Handler of LWX. MAC Handler deals with the tasks of IEEE 802.16 MAC layer including ranging, packet data unit (PDU) generation, call admission control (CAC), bandwidth allocation, etc. PHY Handler handles the functionalities of IEEE 802.16 wireless signal transmission such as modulation and signal coding. LWX Otc1 Script Transformer is designed to translate the designed
simulation script settings into each component within LWX. LWX Simulation Log Generator is used to record the simulation process for some special designs of IEEE 802.16 standard such as the content of DL/UL-MAP message and the structure of each allocated burst that original ns2 simulation log tracer does not support.

The basic workflow of LWX is described as follows:
(1) The user designs his/her simulation scenario and writes the corresponding simulation script.
(2) LWX Ootcl Script Transformer translates the IEEE 802.16 environment settings assigned by the user into each component of LWX, and ns2 script transformer also translates the designed script and triggers ns2 to generate the specified traffic.
(3) When the generated traffics are sent into LWX from ns2, the Traffic Handler aggregates these traffics and maps them to the corresponding IEEE 802.16 connections.
(4) MAC Handler manages these connections according to the functionalities of IEEE 802.16 MAC layer such as generating PDU and DL/UL-MAP, allocating two-dimensional bandwidth, and ranging.
(5) PHY Handler transmits each packet to its next hop.
(6) The node of next hop of the packet moves the packet from PHY Handler to MAC Handler.
(7) MAC Handler deals with the packet from the PHY Handler and then moves it to the corresponding agent of ns2.
(8) LWX Simulation Log Generator and ns2 Log Tracer record the simulation process on the corresponding simulation log files.

3.2. The Core Design of LWX

The core of LWX is MAC Handler. Before developing MAC Handler, we have to register the IEEE 802.16 packet format within the original ns2 system by modifying packet.h and agent.cc. Furthermore, in order to integrate LWX into ns2, three classes within ns2 are used by LWX for inheritance including TclClass, MAC, and Hander class.

For the IEEE 802.16 network environment, several classes are created. hdr_lwx class specifies the IEEE 802.16 packet header and other attributes for simulation usage. LWX_Node and LWX_Node_Info class describe each node’s properties, such as IP address, node type, position, etc. Connection and LWX_Relay_Link_Info class define the IEEE 802.16 connections and the relative attributions such as CINR and QoS parameters. LWX_Wimax_Frame, LWX_Wimax_Zone, Area_Info, and Zone_Unit_Info class present the structure and attributions of IEEE 802.16 frame, zone, burst, and slot, respectively.

LWX class is the major part of MAC Handler that takes complete charge of the functionalities of IEEE 802.16 MAC layer. LWX_Manag() is responsible for managing IEEE 802.16 connections and nodes including ranging, call admission control, connection classifying, and MAC control and data packet management. In addition, to deal with the network traffic, LWX_Manag() uses recv() to receive packet from the Traffic Handler and PHY Handler, manages IEEE 802.16 bandwidth resource by bandwidth_allocation(), and transmits packet to PHY Handler by send_down() and to Traffic Handler by send_up() according to the allocated bandwidth.

There are two components involved in bandwidth allocation named Bandwidth_Scheduler and Burst_Constructur class. Bandwidth_Scheduler is in charge of allocating number of slots to each connection based on its requested bandwidth and QoS. Additionally, it should also decide whether the packet passes RS or it does not in IEEE 802.16j multi-hop relay network. In LWX, we provide three bandwidth scheduling algorithms including Strict Priority, Round Robin, and Simple Relay, where Strict Priority and Round Robin do not consider relay path and Simple Relay does. For Strict Priority, the bandwidth is dispensed in turn to the UGS, erTPS, rTPS, nrtPS, and BE connections. Regarding Round Robin, the bandwidth is equally allocated to each connection. About Simple Relay, we first select the path with best modulation coding rate that each connection owns, and then we adopt round robin discipline to schedule the bandwidth. Burst_Constructur class takes responsible for constructing each connection’s burst, which is a two-dimensional bandwidth, according to its allocated number of slot and channel quality. In LWX, we implement Raster [6] to construct downlink burst and uplink burst.

In order to let the user to manage and control the IEEE 802.16 simulation environment, LWX_Tcl class provides a friendly interface to manage the simulation environment. The user can just only set the desired simulation environment parameters in his/her simulation script, and then LWX_Tcl will bind these settings to the corresponding LWX simulation environment variables. Furthermore, in order to observe what performance the IEEE 802.16 network...
systems will be achieved by adopting and combining different relay path selection, bandwidth scheduling, and burst constructing algorithms, the user can merely modify the simulation script settings to "plug-and-play" different algorithms resident in LWX, and then the dynamic binding design of LWX will automatically switch and adopt the assigned algorithms without modifying LWX and recompiling overall ns2 system which requires tremendous time.

LWX also supplies the complete sample source codes and rich simulation scenario scripts of four implemented algorithms about relay path selection algorithms, bandwidth scheduling, and burst constructing, to help the IEEE 802.16 researchers to understand the software architecture and the workflow of LWX. In advance, by the object-oriented design of LWX and the abundant samples, the researchers can implement and integrate their algorithms into LWX easily and quickly for further researching purposes.

4. Evaluating LWX

In this section, we design a simulation scenario to evaluate LWX. In this simulation, the performances of three bandwidth scheduling algorithms including Strict Priority, Round Robin, and Simple Relay, are compared in terms of peak system rate and $L_{\text{max}}$ violation ratio of rtPS connections. The peak system rate means the maximum transmission rate of downlink and uplink within the IEEE 802.16 network. The $L_{\text{max}}$ violation ratio of rtPS connections is the percentages that the packet delay of rtPS connections in DL or UL violates its $L_{\text{max}}$. For the burst constructing algorithm, we adopt Raster to construct the downlink and uplink burst.

4.1. Simulation Environment

The IEEE 802.16 simulation environment parameters are shown in Table 1. The access bandwidth is used for data transmission directly to or from the SS, and the relay bandwidth is used for relay packets between the RS and BS. The simulation topology, as shown in Figure 3, contains one BS, two RSs, and two SSs.

Table 2 shows the connection settings with regard
to this simulation scenario. The connections from the BS to the SS 1 are downlink connections and the ones from the SS 2 to the BS are uplink connections. The downlink transmission can be relayed by RS 1, and the uplink transmission can be relayed by RS 2. The packet interarrival time of the connections belonging to the service classes of ertPS, rtPS, nrtPS, and BE follows the Exponential distribution. UGS connections generate the traffic in constant bit rate (CBR). During the simulation, different modulation coding rates are adopted according to each connection’s CINR, including 64QAM3/4, 64AQM2/3, 16QAM3/4, 16QAM1/2, QPSK3/4, and QPSK1/2.

To evaluate the performance of three bandwidth scheduling algorithms in different traffic loads, the simulation varies number of BE connections from one to ten in DL and UL.

4.2. Simulation Results

Figure 4 shows the simulation results of three bandwidth scheduling algorithms with the Raster burst constructing algorithm. In Figure 4 (a), Simple Relay obvious outperforms Round Robin and Strict Priority on system peak rate by at most 4 Mbps when the network is highly congestion. This is because Simple Relay can redirect packet to the relay links of which the modulation coding rate is better than that of the access links. As shown in Figure 4 (b), the rtPS $L_{max}$ violation ratio of Round Robin is over 80% and that of other algorithms are 0% when the traffic load is heavy. This is due to that Round Robin doesn’t consider the QoS demands of the rtPS connections, and Strict Priority satisfies the QoS of each connection in turn of its belonging service class, that are UGS, ertPS, rtPS, nrtPS, and BE. With regard to Simple Relay, the bandwidth can satisfy each connection’s requested bandwidth because the modulation coding rate of the paths passing RSs are better than the path directly between the BS and the SS.

5. Conclusions

This paper presents a detailed design and implementation of the IEEE 802.16 simulator, LWX. The contributions of LWX are 1) to support QoS, OFDMA, and multi-hop relay, 2) to provide a dynamic binding mechanism for users to “plug-and-play” different bandwidth allocation algorithms, and 3) to supply the complete sample source codes and rich simulation scripts. We hope that LWX can benefit academic researchers and industrial developers to verify their designed algorithms and mechanism more quickly and easily.

6. References