Evaluating Self Clocked Fair and Weighted Fair Queuing in WiMax

Bharti¹, Yudhvir Singh², Dheer Dhwaj Barak³
¹²³Research Scholar, CSE Department, UIET, MDU Rohtak (Haryana)
²Associate Professor, CSE Department, UIET, MDU Rohtak (Haryana)
dr.yudhvirs@gmail.com

Abstract
WiMAX is one of the hottest broadband wireless technologies around today. WiMAX systems are expected to deliver broadband access services to residential and enterprise customers in an economical way. WiMAX is an industry trade organization formed by leading communications component and equipment companies to promote and certify compatibility and interoperability of broadband wireless access equipment that conforms to the IEEE 802.16 and ETSI HIPERMAN standards. WiMAX (Worldwide Interoperability for Microwave Access) is expected to arise as the main Broadband Wireless Access (BWA) technology providing voice, data and video services with different type of QoS (Quality of Service). Although different type of QoS classes had been defined by the IEEE 802.16 standard, the scheduling architecture is left to be vendor specific. Designing an efficient scheduling algorithm provide high throughput and minimum delay is challenging for system developers. In this work, a detailed simulation study was carried out for scheduling algorithms such as WFQ and SCF, analyzing and evaluating the performance of each scheduler to support the different QoS classes. The simulation is carried out via the QualNet 6.1 simulator evaluation version and the results show that effective scheduling algorithm can provide high service standards to support the QoS required by different type of traffic as well as different type of user.

Keywords: Scheduling algorithms, SCF, WFQ, WiMax, QoS, QualNet.

1. Introduction
Broadband wireless access (BWA) systems, are very flexible and easily deployable high-speed communication systems. BWA systems complement existing last mile wired networks such as cable modem and xDSL[5]. The objective is to have an efficient use of radio resources while serving different types of data flows. These flows can have different constraints such as minimum traffic rate, maximum latency, and tolerated jitter. IEEE 802.16 standards group has been developing a set of standards for broadband (high-speed) wireless access (BWA) in a metropolitan area. Since 2001, a number of variants of these standards have been issued and are still being developed. Like any other standards, these specifications are also a compromise of several competing proposals and contain numerous optional features and mechanisms. This research paper aimed to investigate the performance of WFQ and SCF scheduling algorithms for different QOS classes.

2. IEEE 802.16 QoS
A basic WiMAX network consists of a base station (BS) and multiple subscriber stations (SSs). The BS schedules the traffic flow, communication between BS and SSs are bidirectional, downlink channel (BS
to SS) is in broadcast mode and uplink channel (SS to BS) is shared by various SSs. The standard supports two type of duplex mode, Time Division Duplex (TDD) and Frequency Division Duplex (FDD). The TDD frame consists of downlink and uplink subframes, the duration and the number of subframe slots are determined by the BS scheduler. The downlink subframe has downlink map (DL map) contains information about the duration of subframes and which time slot belongs to a particular SS as the downlink channel and uplink map (UL map) consists of information element (IE) which includes transmission opportunities [2].

A. MAC-Layer Overview
The WiMAX MAC layer provides an interface between the higher transport layers and the physical layer. It takes service data units (MSDUs) packets from the upper layer and organizes them into MAC protocol data units (MPDUs) for sent transmission and vice versa for received transmission. Its design includes a convergence sub layer (CS) that can interface with a variety of higher-layer protocols, such as ATM, TDM, Ethernet, IP, and any future protocol. In addition to providing mapping to and from higher layers, the CS supports MSDU header suppression to reduce the higher layer overheads [2].

B. Quality of Service
Supporting QoS is a fundamental part of the WiMAX MAC layer design. WiMAX borrows some of the basic ideas behind its QoS design from the Data Over Cable Service Interface Specification (DOCSIS) cable modem standard. In a MAC connection oriented architecture, data in between BSs and SSs is transmitted in the context of connection. Each connection is identified in the (MPDU) by a connection identifier (CID) which also provides a mapping to a service flow identifier (SFID). SFID is an important concept in the MAC layer in the standard, which provides a mapping, to the QoS parameters for a particular data entity. WiMAX defines five QoS classes. The QoS requirements for each class are usually mapped into certain bandwidth allocation depending on class specification. There are five levels of QoS at MAC level: [3,6]

a. Unsolicited Grant Services(UGS)
b. Real-time Polling Services(rtPS)
c. Extended real-time Polling Services
d. Non real-time Polling Services
e. Best Effort Services(BE)

a. Unsolicited Grant Services(UGS)
Which is designed to support real-time data streams consisting of fixed-size data packets issued at periodic intervals and services which require Constant Bit Rate (CBR). The BS provides fixed-size data grants at periodic intervals, like the case in E1 and VOIP without silence suppression [2]

b. Real-Time Polling Service (rtPS)
Which is designed to support real-time data streams consisting of variable-sized data packets that are issued at periodic intervals. The BS provides periodic unicast (uplink) request opportunities, like the case in MPEG video transmission [2].

c. Extended real time Polling Services
Which is suitable for variable rate real time applications that have data rate and delay requirements, like the case in VOIP without silence suppression. The IEEE 802.16e standard indicates that rtPS is built upon the efficiency of both UGS and rtPS. The BS provides unicast grants in an unsolicited manner like in UGS [2]

d. Non-Real-Time Polling Service
Which is designed to support delay tolerant data streams consisting of variable size data packets for which a minimum data rate is required, like the case in FTP traffic. The BS provides unicast uplink request polls on a regular basis, which guarantees that the service flow receives request opportunities even during network congestion [2].

e. Best Effort (BE) service
Which is designed to support data streams for which no minimum service guarantees are required, like the case in HTTP traffic. The BS does not have any unicast uplink request polling obligation for BE SSs. Therefore, a long period can run without transmitting any BE packets [2].

3. Scheduling Algorithms
Scheduling algorithms are responsible for Distributing resources among all users in the network, and provide them with a higher QoS. Users request different classes of service that may have different requirements (such as bandwidth and delay), so the main goal of any scheduling algorithm is to maximize the network utilization and achieve fairness among all users.
3.1 Self-Clocked-Fair (SCF):
It is an efficient queuing scheme which satisfies the quality of services (QoSs) in broadband implementation. The algorithm is based on the concept of virtual time that adopts the concept of an internally generated virtual time as the index of work in progress. It links virtual time to the work progress in the fluid-flow fair queuing (FFQ). As virtual time function is involved in determining the order of which packet should be served next, the virtual time that is produced depends very much on the progress of work in the actual packet based queuing system. This scheme is efficient for the internal generation of virtual time as it involves negligible overhead. This is because virtual time is easily computed from the packet situated at the head of the queue. In addition, the SCFQ algorithm can accomplish easier implementation and it can maintain the fairness attribute in virtual time function. [1]

Advantages:
- Simplify the complexity of calculating the finish time in a corresponding GPS system. The decrease in complexity result in worse case delay and the delay increase with the number of service classes.
- Efficient for the internal generation of virtual time as it involve negligible overhead.

Disadvantage:
- Virtual time that is produced depends very much on the progress of work in the actual packet based queuing system.

3.2 Weighted-Fair-Queuing (WFQ):
This algorithm is employed for uplink traffic in WiMAX with different size packets. As it caters to different size packets, it emphasizes on providing fair scheduling for the different flows by assigning finish times to the packets. The finish times are based on the size and weight of the packets. In general, the WFQ algorithm outperforms the WRR due to variable size packets. However, the weaknesses of WFQ algorithm are, the start time of a packet is not taken into consideration, and it can lower the scheduler system if many packets occur in the priority region WFQ support for flows with different bandwidth requirements by giving each queue a weight that assigns it different percentages of output port bandwidth. [1]

Advantages:
- Provide protection to each service class by ensuring a minimum level of output port bandwidth independent of behavior of other service classes.

Disadvantages:
- Computational complexity impacts the scalability of WFQ when attempting to support a large number of service classes of high speed interfaces.

4. Simulation Model

The overall goal of this simulation study is to evaluate the performance of different existing scheduling algorithms in Mobile WiMAX environment. The simulations have been performed using QualNet 6.1 evaluation version.
Simulation run simulates 10 minutes of real operation of the network which took around 1/2 minute as simulation time. This due to high traffic rate used to enable us measure the performance of the algorithm under high traffic volume. The important parameters used to configure the PHY and MAC layers are summarized in (table I).

### Table 1: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antenna Height (meter)</td>
<td>5</td>
</tr>
<tr>
<td>Antenna Model</td>
<td>Omni directional</td>
</tr>
<tr>
<td>Interval between Packets</td>
<td>1 sec</td>
</tr>
<tr>
<td>Channel Frequency</td>
<td>2.4 GHz</td>
</tr>
<tr>
<td>Item Size</td>
<td>512</td>
</tr>
<tr>
<td>Node Placement</td>
<td>Random</td>
</tr>
<tr>
<td>Number of CBR</td>
<td>20</td>
</tr>
<tr>
<td>Precedence Value</td>
<td>0,1,3,4,5</td>
</tr>
<tr>
<td>Radio Type</td>
<td>802.16 Radio</td>
</tr>
<tr>
<td>Scheduling Algorithm Evaluated</td>
<td>WFQ, SCF</td>
</tr>
<tr>
<td>Simulation Grid Size</td>
<td>2000 m × 2000 m</td>
</tr>
<tr>
<td>Simulation Time</td>
<td>10 min</td>
</tr>
<tr>
<td>FFT Size</td>
<td>2048</td>
</tr>
<tr>
<td>Simulator</td>
<td>QualNet 6.1</td>
</tr>
</tbody>
</table>
We considered scenario in our simulation; For the scenario we have 35 nodes randomly located in a space of 2000 m × 2000 m. The IEEE 802.16 is used as a MAC layer communication protocol. In the application layer, the nodes communicate using Constant Bit Rate (CBR) traffic generators over UDP with random source/destination pairs. The CBR is a model for sending data packets from sources to destinations. In our scenario we have taken 20 CBR traffic.

To measure the performance of our algorithms, we studied the effect of mobility for different QoS on WFQ and SCF the simulation. Table1 illustrates the simulation parameters depending on the related work.

Table 1 illustrates the simulation parameters depending on the related work.

Table 2: IP output queue weight

<table>
<thead>
<tr>
<th>IP Output Queues</th>
<th>Weight Assigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index 0</td>
<td>0.1</td>
</tr>
<tr>
<td>Index 1</td>
<td>0.1</td>
</tr>
<tr>
<td>Index 2</td>
<td>0.1</td>
</tr>
<tr>
<td>Index 3</td>
<td>0.1</td>
</tr>
<tr>
<td>Index 4</td>
<td>0.1</td>
</tr>
<tr>
<td>Index 5</td>
<td>0.1</td>
</tr>
<tr>
<td>Index 6</td>
<td>0.2</td>
</tr>
<tr>
<td>Index 7</td>
<td>0.2</td>
</tr>
</tbody>
</table>

4.1. Performance Metrics
For the purpose of evaluating scheduling techniques and comparing their performance, we used some common performance metrics: Average Jitter, Average end to end Delay, and Average Throughput.

• **Average Jitter:** This performance metric represents the inter-packet arrival time to the receiver and is required to be reasonably stable by the real-time applications.

• **Average End-to-End Delay:** This performance metric represents the average delay between the time when the data packet was originated at the source node and the time it reaches the destination node. The end to end delay metric includes delays due to route discovery, queuing and transmissions at the MAC level. The value is expressed in seconds.

• **Average Throughput:** Average throughput represents the amount of data transmitted by user per unit time. The value is expressed in Kbps.

To simulate different QoS types a mapping with different precedence values are used as shown in Table 3.
Table 3: MAC Layer Service Flow Mapping Mac Layer Service Precedence

<table>
<thead>
<tr>
<th>Qos</th>
<th>Precedence Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BE</td>
<td>0</td>
</tr>
<tr>
<td>nrtPS</td>
<td>1, 2, 6</td>
</tr>
<tr>
<td>rtPS</td>
<td>3</td>
</tr>
<tr>
<td>ertPS</td>
<td>4</td>
</tr>
<tr>
<td>UGS</td>
<td>5, 7</td>
</tr>
</tbody>
</table>

5. Results

In this section, the output of simulation is shown and analyzed, the results shown in this section is measured in high congested environment. Analysis is based on above given scenario and parameter used, that are common for both scheduling algorithms. Calculations are done for parameter like delay (sec), throughput (bits/sec) and jitter (sec).

Figure 2 shows that for best effort Qos, both scheduling algorithm generate approx. same amount of delay at mobility 0 and 20. But as the mobility increases WFQ outperforms other SCF. Finally at mobility 80 and 100, both algorithm produce approx same delay.
Fig 3: Throughput v/s Speed

Figure 3 shows that for the best effort QoS, throughput of WFQ outperform SCF. At mobility 60 and 100 both scheduling algo. have same throughput.

Fig 4: Jitter v/s Speed

Figure 4 shows that initially both scheduling generate same amount of jitter. But at mobility 40 and 60, WFQ outperforms SCF in performance. Finally, at mobility 80 and 100 both scheduling have approx same jitter.
Figure 5 shows that for nrtPS QoS, both scheduling algorithms generate approx. same amount of delay at mobility 0, 20 and 40. But at the mobility 60, WFQ outperforms SCF algorithm. Finally at mobility 100, both algorithms produce approx. same delay.

Figure 6 shows that for the nrtPS QoS, throughput of both algorithms is approx. same at mobility 0, 20, 60, 80, and 100. But at mobility 40, WFQ outperforms SCF scheduling algorithm.
Figure 7 shows that for nrtPS QoS, initially at mobility 0, 20 and 40, both scheduling algorithms generate same amount of jitter. But at mobility 60, WFQ outperforms SCF in performance. Finally, at mobility 80 and 100 both scheduling algorithms have approx same jitter.

Figure 8 shows that for rtPS QoS, both scheduling algorithm generate approx. same amount of delay at mobility 0, 20 and 40. But as the mobility increases WFQ has highest delay at mobility 60. Finally at mobility 100, both algorithms produce approx same delay.
Figure 9 shows that for the rtPS QoS, throughput of both scheduling algorithms are approx. same. But at mobility 0, 60 and 100, WFQ outperforms WFQ scheduling. Results show that both scheduling algorithms gives better performance at low mobility. Also, as the mobility increases, performance decreases.

Figure 10 shows that for rtPS QoS, initially at mobility 0, 20 and 40, both scheduling algorithms generate same amount of jitter. But at mobility 60, SCF outperform WFQ in performance. Finally, at mobility 80 and 100 both scheduling algorithms have approx same jitter.
Figure 11 shows that for ertPS QoS, both scheduling algorithms generate approx. same amount of delay at mobility 0, 20 and 40. But as the mobility increases, WFQ generate highest delay. Finally at mobility 80 and 100, both algorithms produce approx same delay.

Figure 12 shows that for the ertPS QoS, throughput of WFQ outperform at mobility 0, 80 and 100, but at mobility 20, 40 and 60 SCF scheduling algo. outperforms WFQ scheduling algorithm.
Figure 13 shows that for ertPS QoS, initially at mobility 0, 20 and 40, SCF scheduling algorithms generate large amount of jitter than other, but at mobility 60, WFQ have highest jitter. Finally, at mobility 80 and 100 both scheduling have approx same jitter.

Figure 14 shows that for UGS QoS, both scheduling algorithm generate approx same amount of throughput at all mobility.
Figure 15 shows that for UGS QoS, both scheduling algorithms generate approx. same amount of delay at all mobility.

Figure 16 shows that for UGS QoS, both scheduling algorithms generate approx. same amount of jitter at all mobility.
6. Conclusion and Future Scope

In this paper, we studied the behaviors of scheduling algorithms namely WFQ and SCF, which has shown the strength of both scheduling algorithms at varying mobility. Various performance metrics are calculated for both scheduling algorithms for each QoS class.

For best effort and nrtPS, SCF shows highest delay and jitter. Also, WFQ outperforms SCF. For rtPS QoS, both scheduling algorithms give approx same performance for all mobility's. For the rtPS QoS, throughput of WFQ outperforms at mobility 0, 80 and 100, but at mobility 20, 40 and 60 SCF scheduling algorithm Outperforms WFQ scheduling algorithm. For UGS QoS, both scheduling algorithms perform same at all mobilities for all performance metrics.

For future studied of scheduling algorithms, we recommend that the performance SP, WFQ and SCF will be studied for number of subnets and using large simulation grid size such as 3000m x3000m and 6000 mx 5000m with large number of subscriber station. Also it is important to consider the number of IP input queues, IP output queues and size of input Queue. Finally number of loads and type of traffic are investigated for these network scenarios.

Reference