An Enhanced RED-Based Weighted Fair Priority Queuing Algorithm for IEEE 802.16 Subscriber Station Scheduler

Serda Kasaci, Sema Oktug
Department of Computer Engineering
Istanbul Technical University
Istanbul, Turkey
e-mail: {kasaci, oktug}@itu.edu.tr

***Abstract*— IEEE 802.16, which is called as Worldwide Interoperability for Microwave Access (WiMAX), is an air interface for Fixed Broadband Wireless Access Systems. In the 802.16 IEEE standard, different service types are introduced, such as Unsolicited Grant Service (UGS), Real-time Polling Service (rtPS), Non-Real-time Polling Service (nrtPS) and Best Effort (BE). Each service type is associated with a set of Quality of Service (QoS) parameters; however WiMAX does not specify how to schedule the granted bandwidth efficiently between these service classes. In this paper, we propose an Enhanced RED-based Weighted Fair Priority Queuing algorithm for Subscriber Stations. The weights are calculated according to the traffic load of the rtPS and nrtPS service classes. Simulation results show that, both rtPS and nrtPS throughputs are improved without starving lower priority service classes.**

Keywords- wimax; scheduling; uplink; GPSS; QoS

I. INTRODUCTION

The IEEE 802.16 standard, widely known as WiMAX, has been developed for Broadband Wireless Access (BWA). The advantages of this standard are easy and low-cost deployment, high speed data rate, last mile wireless access, and QoS support for multimedia applications [1]. The standard defines two possible network topologies, such as *Point-to-Multipoint (PMP)* and *Mesh* Networks. In the PMP networks, communication between Subscriber Stations (SSs) is possible only through a Base Station (BS). In the mesh mode, SSs can communicate with each other directly. In this paper, we employ PMP topology.

The standard IEEE 802.16 defines the physical layer and the *MAC (Medium Access Control)* layer. The main purpose of the MAC protocol is to share radio channel resources among multiple accesses of different users. As the MAC protocol is connection-oriented, all data transmission takes place in connections, even for connectionless packets. The MAC layer contains three sublayers such as: Convergence Sublayer (CS), Common Part Sublayer (CPS), and the Security Sublayer. CS accepts Protocol Data Units (PDUs) from higher layers. The MAC SDUs are classified and mapped into appropriate Connection IDentifiers (CIDs) and they are transmitted to CPS by CS. CPS is responsible for fragmentation and segmentation of each MAC SDU into MAC PDUs, system access, bandwidth allocation, connection maintenance, QoS control, and scheduling transmission. The Security Sublayer is responsible for security, authentication, and encryption.

The PHY Layer establishes the physical connection between uplink and downlink directions. This layer is responsible for transmission of the bit sequences. There are two dublexing techniques for PHY layer of downlink and uplink such as; Frequency Division Duplex (FDD) and Time Division Duplex (TDD). FDD requires two distinct channels to transmit downlink sub-frame and uplink sub-frame at the same time slot. In TDD, downlink (DL) and uplink (UL) subframes share the same frequency; but they take place at different times. DL Subframe has DL-MAP, UL-MAP and DL PHY PDUs. The DL-MAP message defines the usage of the downlink intervals. The UL-MAP defines the uplink usage in terms of the offset of the burst relative to the Allocation Start Time [2]. UL Subframe contains contention slot for initial ranging, contention slot for bandwidth requests and UL PHY PDUs from SSs. Via Initial Ranging IE, BS provides an interval for new stations to join to the network. Ranging Request (RNG-REQ) packets are used in this interval. Via Request IE, BS specifies an uplink interval which can be used by SS to send a bandwidth requests using contention slots.

There are four service types defined in IEEE.802.16-2004; Unsolicited Grant Service (UGS), real-Time Polling Service (rtPS), non-real-time Polling Service (nrtPS), and Best Effort (BE). In the 802.16e standard [3], a new service type, called extended real time Polling Service (ertPS), has been added. However, it is out of the scope of this paper.

UGS supports Constant Bit Rate (CBR) flows for real-time applications; such as VoIP without silence suppression or E1/T1 data streams. The BS allocates fixed sized data grants at periodic intervals based on the Maximum Sustained Traffic Rate of the service flow. The overhead and latency of SS requests are eliminated for UGS connections. However, UGS is more expensive than other service types.

Variable Bit Rate (VBR) flows, which have variable packet length and periodic packet intervals, such as Moving Pictures Expert Group video, are supported by rtPS. BS provides unicast request opportunities to SSs periodically.

Variable-sized packets, which are delay-tolerant data streams, such as File Transfer Protocol (FTP), are supported by nrtPS. Therefore the minimum data rate is required for this service. BS provides unicast request opportunities periodically as in the rtPS service, so this will guarantee data granting during network congestion. In addition to this, SSs can use contention request mechanism.

Best Effort is designed for best-effort traffic such as HTTP, and this service does not have any minimum service guarantee. SS can use contention request opportunities to send any bandwidth request.

A bandwidth request may be a standalone BW request header or it may come as a Piggyback Request. Some policies are used to send the BW such as unicast or multicast polling, using contention request slots or setting Poll-Me bit. There are two types of BW Requests: incremental and aggregate. BW is requested on a CID basis, but bandwidth grants are allocated on an SS basis. IEEE 802.16 MAC accommodates two modes of SS, differentiated by their ability to accept bandwidth grants simply for a connection or for the SS as a whole. In Grant Per Connection (GPC) mode, bandwidth is granted to a connection, so SS can use this grant for this connection only. In the Grant Per Subscriber (GPSS) mode, the BS grants bandwidth to an SS as an aggregate of grants in response to per connection requests from the SS. Then the SS distributes bandwidth among its connections, with respect to their QoS requirements. Therefore, the GPSS mode is more complex than the GPC mode.

A scheduling algorithm has to determine the allocation of the bandwidth among the users and their transmission order. QoS requirements of the users need to be satisfied while utilizing the available bandwidth efficiently [4]. There are two types of schedulers: the SS Scheduler and the BS scheduler. The SS Scheduler is more complicated in the GPSS mode, as the algorithm which works in the SS scheduler distributes the granted bandwidth between its connections [5]. As the WiMAX standard does not specify how to efficiently schedule traffic to fulfill QoS requirements, a lot of research has been done on this topic. Several works have introduced algorithms for the schedulers in the Base Station (BS) and the Subscriber Station (SS).

In this paper, we focus on the GPSS type of SS scheduler and their performance. Strict Priority (SP), Weighted Fair Priority Queuing (WFPQ), and RED-based Deficit Fair Priority Queuing (DFPQ) are investigated. We propose an enhanced RED-based WFPQ algorithm to increase both rtPS and nrtPS throughput. This algorithm is called as Enhanced RED-based Weighted Fair Priority Queuing and has a dynamic structure while granting bandwidth between service classes. This algorithm takes the packet size information of rtPS and nrtPS, and then calculates the weights of the service flows based on the RED technique.

The rest of the paper is organized as follows. Section II presents previously introduced scheduling algorithms for SS Schedulers in PMP WiMAX networks. The proposed scheduling algorithm is described in Section III. Simulation results are shown and discussed in Section IV. Section V concludes the paper by giving future directions.

II. ALGORITHMS FOR SS SCHEDULER IN 802.16 NETWORKS

Several scheduling algorithms have been proposed for SS schedulers in PMP WiMAX networks to improve the performance of the system; such as Strict Priority, Weighted Fair Priority Queuing, and RED-based Deficit Fair Priority Queuing.

A. *Strict Priority:*

Bandwidth is allocated for rtPS service flows first, then bandwidth is allocated for nrtPS service flows, and finally the remaining bandwidth is allocated for BE service flows. Consequently, under heavy rtPS traffic load, nrtPS and BE service flows may starve. Strict Priority scheduling does not guarantee the QoS requirements of the traffic that comes from lower priority service classes.

B. *Weighted Fair Priority Queuing*

WFPQ scheduling is a generalization of Fair Queuing. WFPQ allows different sessions to have different service shares. A link data rate (*R*), is serviced for the active data flows (*N*). The data rate of session *j* is calculated as follows:

$$R_j = \frac{R \times w_j}{\sum_{i=1}^{N} w_i} \qquad (1)$$

where w_j represents the weight assigned to session *j*.

According to (1), the available bandwidth is shared between the service types in the SS Scheduler. Therefore, we need to define the weights for service types efficiently. For example, as the priority of rtPS is higher than nrtPS, rtPS needs to be given a higher weight than nrtPS.

C. *RED-based Deficit Fair Priority Queuing*

Chen et al. proposed the Deficit Fair Priority Queuing based scheduler for bandwidth allocation among the service classes of WiMAX networks [6]. It uses Deficit Counters (DCs) for rtPS, nrtPS, and BE. In Fig. 1, the DC for rtPS service class is adaptively calculated according to RED technique. If the current packet length of the rtPS queue ($QL_{current}$) is less than $QL_{threshold1}$, the DC value will be equal to DC_{min}. If the $QL_{current}$ is between $QL_{threshold1}$ and $QL_{threshold2}$, DC will be equal to $DC_{dynamic}$. The $DC_{dynamic}$ is calculated using (2). If the $QL_{current}$ is more than $QL_{threshold2}$, DC equals to $DC_{max.}$

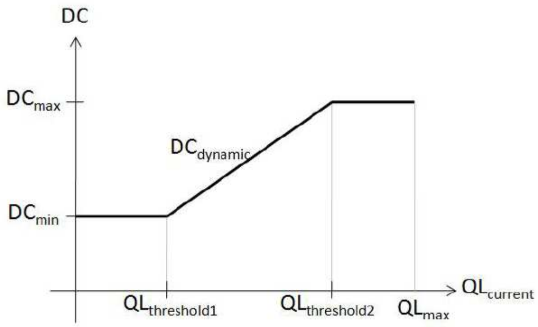

Figure 1. RED-based Deficit Fair Priority Queuing [6]

$$\begin{aligned} DC_{\min} &= Q_{rtPS} \\ DC_{dynamic} &= Q_{rtPS} + \frac{QL_{current} - QL_{threshold\,1}}{QL_{threshold\,2} - QL_{threshold\,1}} \times Q_{rtPS} \\ DC_{\max} &= 2 \times Q_{rtPS} \end{aligned} \quad (2)$$

Following the transmission of rtPS packets, nrtPS packets will be transmitted. If there is no rtPS or nrtPS packet left, scheduler transmits BE packets.

III. PROPOSED ALGORITHM

In this paper, we propose a new SS Scheduler algorithm which is called Enhanced RED-based Weighted Fair Priority Queuing and derived from RED-based Weighted Fair Priority Queuing (WFPQ). RED-based WFPQ is simple than RED-based DFPQ, as we do not deal with deficit counters; we only determine weights for the service types.

A. *RED-Based Weighted Fair Priority Queuing*

When the rtPS queue length is lower than $QL\_Th_{rtps\_min}$, W_{rtps_min} is assigned for the weight of the rtPS service flow. When the rtPS queue length is higher than $QL\_Th_{rtps\_max}$, W_{rtps_max} is assigned to rtPS. When the rtPS queue length is between $QL\_Th_{rtps\_min}$ and $QL\_Th_{rtps\_max}$, the rtPS weight changes dynamically according to the rtPS queue length. The slope of W_{rtps} is calculated according to (3). Equation (5) represents for the weight assignment of rtPS.

$$m_{rtps} = \frac{W_{rtps\_max} - W_{rtps\_min}}{QL\_Th_{rtps\_max} - QL\_Th_{rtps\_min}} \quad (3)$$

According to Fig. 2, W_{rtps} is calculated at the beginning of every frame by using the diagram. The rest of the available weights are distributed between nrtPS and BE flows according to their weights (W_{nrtps} and W_{BE}).

B. *Enhanced RED-Based Weighted Fair Priority Queuing*

In Enhanced RED-based WFPQ algorithm, we do not define static weight for nrtPS. We apply the dynamic weight assignment of RED-based WFPQ algorithm to nrtPS service types. As the dynamic weight assignment is used for both rtPS and nrtPS, we call this algorithm “Enhanced RED-based WFPQ” algorithm.

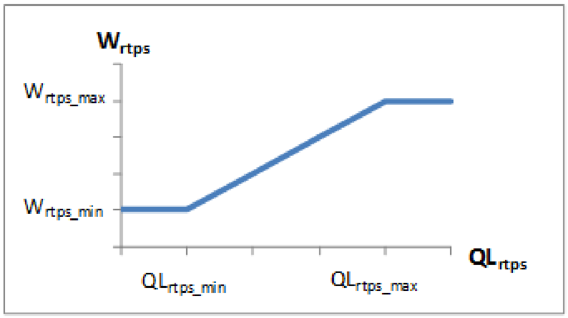

Figure 2. RED-based weights for rtPS service class

TABLE I. RTPS QUEUE LENGTH CONDITIONS

Condition-1	[$QL_{rtps} < QL\_Th_{rtps\_min}$]
Condition-2	[$QL\_Th_{rtps\_min} < QL_{rtps} < QL\_Th_{rtps\_max}$]
Condition-3	[$QL_{rtps} > QL\_Th_{rtps\_max}$]

The weight assignment of the rtPS service type is the same as in RED-based WFPQ. As the nrtPS weight depends on the variation of the rtPS weight (total weight is distributed between all service types), we need to consider three conditions while determining the nrtPS weight. The conditions are given in Table I. rtPS values are determined based on the conditions; therefore nrtPS weight characteristics are dynamic and depend on rtPS weight.

Condition-1

In Fig. 2, when the rtPS queue length (QL_{rtps}) is lower than $QL\_Th_{rtps\_min}$, the rtPS weight is set to the predefined W_{rtps_min} value. Weight assignment of nrtPS is represented in (6). When nrtPS queue length (QL_{nrtps}) is lower than the minimum threshold of nrtPS, W_{nrtps_min} is assigned as the weight of nrtPS. When QL_{nrtps} is between minimum and maximum threshold values, the nrtPS weight varies dynamically. When QL_{nrtps} is higher than the maximum threshold of nrtPS, W_{nrtps_max} is assigned as the nrtPS weight. Fig. 3 represents for the RED-based weights for nrtPS service class. In each condition, weights for BE service types are calculated according to (4).

$$W_{BE} = W_{total} - W_{rtps} - W_{nrtps} \quad (4)$$

$$W_{rtps} = \begin{cases} W_{rtps\_min} & if\ \left(QL_{rtps} \le QL\_Th_{rtps\_min}\right) \\ W_{rtps\_min} + m_{rtps} \times \left(QL_{rtps} - QL\_Th_{rtps\_min}\right) & if\ \left(QL\_Th_{rtps\_min} < QL_{rtps} < QL\_Th_{rtps\_max}\right) \\ W_{rtps\_max} & if\ \left(QL_{rtps} \ge QL\_Th_{rtps\_max}\right) \end{cases} \quad (5)$$

ISBN: 978-1-61208-103-8

$$W_{nrtps} = \begin{cases} W_{nrtps\_min} & if\left(QL_{nrtps} \le QL\_Th_{nrtps\_min}\right) \\ W_{n\,rtps\_min} + m_{n\,rtps} \times \left(QL_{n\,rtps} - QL\_Th_{n\,rtps\_min}\right) & if\left(QL\_Th_{nrtps\_min} < QL_{nrtps} < QL\_Th_{nrtps\_max}\right) \\ W_{nrtps\_max} & if\left(QL_{nrtps} \ge QL\_Th_{nrtps\_max}\right) \end{cases} \quad (6)$$

$$W_{nrtps} = \begin{cases} W_{nrtps\_min} = \left(W_{Total} - W_{BE\_min} - \left(QL\_Th_{nrtps\_max} - QL\_Th_{nrtps\_min}\right) \times m_{n\,rtps} - W_{rtps}\right) & if\left(QL_{nrtps} \le QL\_Th_{nrtps\_min}\right) \\ W_{n\,rtps\_min} + m_{n\,rtps} \times \left(QL_{n\,rtps} - QL\_Th_{n\,rtps\_min}\right) & if\ \left(QL\_Th_{nrtps\_min} < QL_{nrtps} < QL\_Th_{nrtps\_max}\right) \\ W_{nrtps\_max} = W_{Total} - W_{rtps} - W_{BE\_min} & if\left(QL_{nrtps} \ge QL\_Th_{nrtps\_max}\right) \end{cases} \quad (7)$$

Condition-2

In Fig. 2, when the rtPS queue length (QL_{rtps}) is lower than $QL\_Th_{rtps\_max}$ and higher than $QL\_Th_{rtps\_min}$, the rtPS weight is set dynamically according to queue length by using (5). Consequently, the available weight, that remains for nrtPS and BE service type, is ($W_{total} - W_{rtps}$). Fig. 3 displays the variation of the nrtPS weight is RED-based, and details are given in (7).

In each condition, weights for BE service types are calculated according to (4). We reserve a little bandwidth for BE flow (W_{BE_min}) to prevent from starving in a congested network.

Condition-3

In Condition-3, rtPS queue length (QL_{rtps}) is higher than $QL\_Th_{rtps\_max}$. Consequently, the rtPS weight is set to $W_{rtps\_max.}$ In this condition, nrtPS and BE weights are statically assigned. The possible maximum value for nrtPS (W_{nrtps_max}) is assigned to W_{nrtps}. The weight of BE is calculated according to (4).

In all three conditions, the maximum values of nrtPS are not the same, as the weight intervals depend on the weight rtPS.

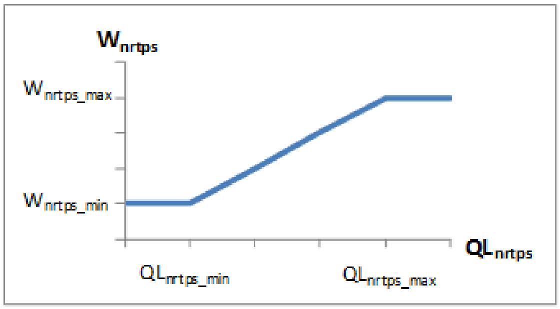

Figure 3. RED-based weights for nrtPS service class

IV. SIMULATION RESULTS

In this paper, the simulations are performed by using IEEE 802.16 WiMAX NIST [7] module which has been developed on NS-2 version 2.29 [8]. We used the WIMAX QoS patch which is designed for NIST WIMAX module [9, 10]. We added nrtPS service class to the patch. The fundamental simulation parameters are shown in Table II. The existing QoS-included WiMAX Patch supports only one connection per subscriber, so we modified the patch to support GPSS mode. We run simulation for throughput analysis 5 times to achieve results with 95% confidence interval.

TABLE II. SIMULATION PARAMETERS

PHY specification	WirelessMAN-OFDM
Frequency Band	5MHz
Antenna Model	Omni Antenna
Antenna Height	1.5 m
Propagation Model	TwoRayGround
Transmit Antenna Gain	1
Transmit Power	0.25 W
Frame Duration	20 ms
Cyclic Prefix	0.025 s
Simulation Duration	100 s
Packet Length	1000 bytes
Frame Structure	TDD

TABLE III. RED-BASED WFPQ SCHEDULER PARAMETERS

Variables	Selected Values
$W_{rtps\ min}$	0.5
$W_{rtps\ max}$	0.7
$QL\_Th_{rtps\ min}$	10 packets
$QL\_Th_{rtps\ max}$	30 packets
$QL_{rtps\ max}$	50 packets
W_{nrtps}	W_{nrtps}
W_{BE}	(2/3) W_{nrtps}
m_{rtps}	0.01
W_{Total}	1

TABLE IV. ENHANCED RED-BASED WFPQ SCHEDULER PARAMETERS IN CONDITION-1

Variables	Selected Values
W_{nrtps_min}	0.35
W_{nrtps_max}	0.45
$QL\_Th_{nrtps\_min}$	10 packets
$QL\_Th_{nrtps\_max}$	30 packets
QL_{nrtps}	50 packets
m_{nrtPS}	0.05
W_{rtps_min}	0.5

The scheduler parameters used throughout the simulations are given in Table III. Fig. 4 shows the behavior of RED-based WFPQ when we use the values in Table III. As we use the same values for rtPS parameters in RED-based WFPQ and Enhanced RED-based WFPQ, their weight graphs are the same. The parameters of Enhanced RED-based WFPQ in Condition-1, which are used for nrtPS weight determination, are chosen as in Table IV.

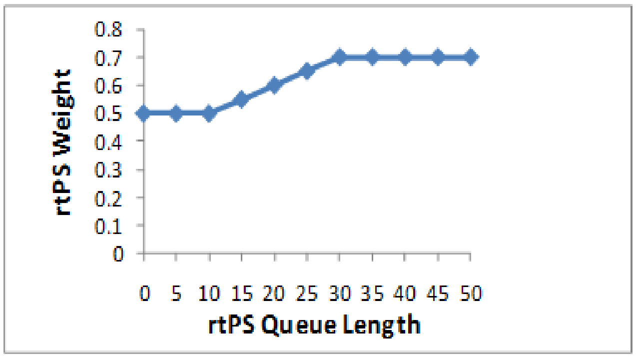

Figure 4. RED-based weights for rtPS service flow

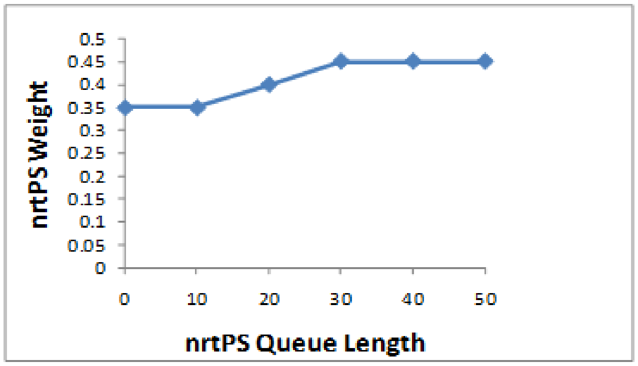

Figure 5. nrtPS weights of Enhanced RED-based WFPQ in Condition-1

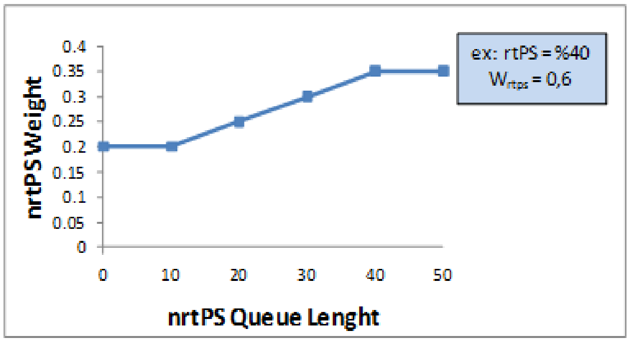

Figure 6. nrtPS weights of Enhanced RED-based WFPQ in Condition-2

Fig. 5 shows the behavior of the algorithm when we use the parameter values in Table V.

Parameters of nrtPS and rtPS, which are used in Condition-2 for Enhanced RED-based WFPQ, are given in Table V. Fig. 6 shows the behavior of the algorithm when we use the parameter values nrtPS as in Table V. In this condition, the diagram depends on rtPS weights. Therefore, to draw a diagram for nrtPS weights, we need to select an rtPS weight value. In our example, we take 0.6 as an example for rtPS.

TABLE V. ENHANCED RED-BASED WFPQ SCHEDULER PARAMETERS IN CONDITION-2

Variables	Selected Values
W_{nrtps_min}	0.2
W_{nrtps_max}	0.35
$QL\_Th_{nrtps\_min}$	10 packets
$QL\_Th_{nrtps\_max}$	40 packets
QL_{nrtps}	50 packets
m_{nrtPS}	0.05
rtPS weight example	0.6
rtPS QL percentage	40%
W_{BE_min}	0.05

TABLE VI. ENHANCED RED-BASED WFPQ SCHEDULER PARAMETERS IN CONDITION-3

Variables	Selected Values
$W_{rtps} = W_{rtps\_max}$	0.7
$W_{nrtps} = W_{nrtps\_max}$	0.25
$W_{BE} = W_{Total} - W_{rtps\_max} - W_{nrtps\_max}$	0.05

Parameters of service flows for Condition-3, are given in Table VI. W_{BE} is calculated according to (4). We need to subtract W_{rtps} and W_{nrtps} from W_{Total}.

We measured throughput of rtPS, nrtPS and BE service class flows. We also calculated the queuing delay, dropped packet percentage and fairness index. We compared the following schedulers with each other:

- Strict Priority Scheduling
- WFPQ Scheduling
- RED-Based WFPQ Scheduling
- Enhanced RED-Based WFPQ Scheduling

In Fig. 7, we consider the rtPS throughput versus increasing rtPS traffic load. Strict Priority scheduling has the maximum throughput level as the algorithm always grants bandwidth for rtPS first, if there is no packet in the rtPS queue and there is available bandwidth left for the SS, then the bandwidth is allocated for the nrtPS service flow. If there are no packets in rtPS and nrtPS queues and there is available bandwidth left for the SS, then the bandwidth is allocated to the BE service flow. In the WFPQ algorithm, as the weights are chosen statically, we cannot increase the throughput of rtPS significantly while increasing rtPS load. RED-based WFPQ and Enhanced RED-based WFPQ have higher rtPS throughput as they can dynamically change the weights of rtPS according to the queue length of the rtPS flow. Initial weights are the same in WFPQ and RED algorithms. However, as rtPS load submission increases, due to higher rtPS traffic, the queue length will also increase. Consequently, RED-based WFPQ and Enhanced RED-based WFPQ yield better performance than WFPQ.

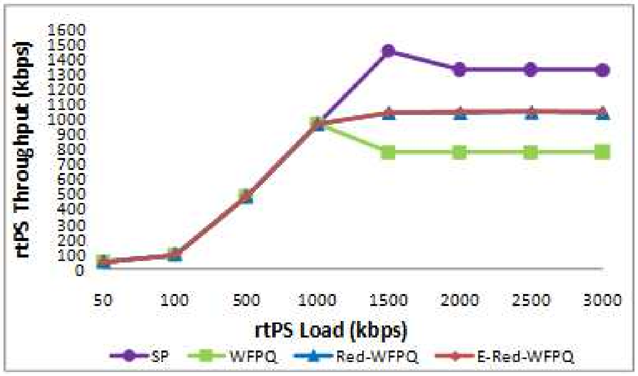

Figure 7. Comparison of rtPS Throughput

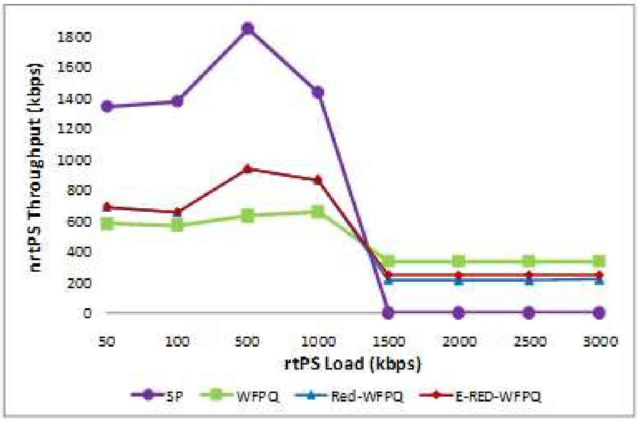

Figure 8. Comparison of nrtPS Throughput

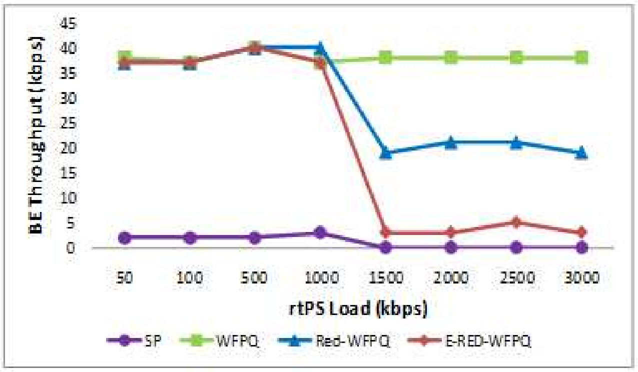

Figure 9. Comparison of BE Throughput

In Fig. 8, for Strict Priority, when rtPS traffic increases significantly, there will be no resource left for nrtPS and BE flows. Therefore, their throughputs will drop to zero. As a result, nrtPS and BE flows may starve under high rtPS traffic. Under high rtPS traffic, WFPQ has the highest nrtPS throughput as it has the maximum nrtPS weights than the others. The main reason is that, WFPQ has lower rtPS load than the others, so it can grant more weight for the nrtPS flow. Enhanced RED-based WFPQ has higher nrtPS throughput than RED-based WFPQ. When the rtPS load is 3000 kbps, Enhanced RED-based WFPQ yields 244 kbps and RED-based WFPQ yields 214 kbps throughput. This means that Enhanced RED-based WFPQ increases the nrtPS throughput as much as 14% over RED-based WFPQ.

In Fig. 9, the WFPQ algorithm yields the best throughput. This is because among all the algorithms, WFPQ assigns the highest weight to BE. However, we do not need to grant bandwidth for BE flows, as they do not have significant QoS requirements. As long as we prevent the starvation of the BE flows in a congested network, we have an acceptable QoS-based system. Therefore, in RED-based WFPQ and Enhanced RED-based WFPQ, we allocate very little weight to BE, so that we can continue serving them. In Strict Priority, the BE users have no chance of being served if the network is congested. Consequently, as rtPS load increases, BE flows cannot transmit their packets, and their throughputs drops to zero beyond 1500 kbps rtPS load.

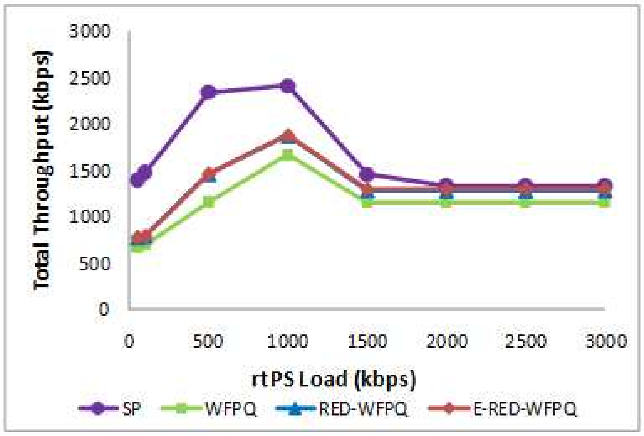

Figure 10. Comparison of Total Throughput

Figure 11. Comparison of rtPS Delay

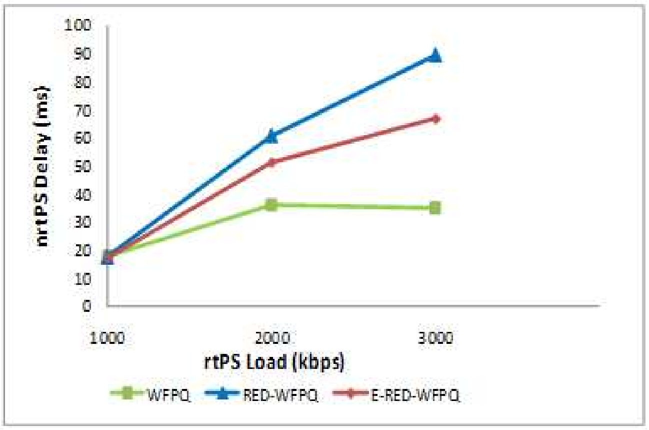

Figure 12. Comparison of nrtPS Delay

According to Fig. 10, Strict Priority scheduling has the maximum total throughput; however, it is not fair and acceptable for a QoS-based system. RED-based WFPQ and Enhanced RED-based WFPQ algorithms yield better throughput than WFPQ. In addition, RED-based WFPQ and Enhanced RED-based WFPQ algorithms yield approximately the same total throughput. The reason is that in our algorithm, Enhanced RED-based WFPQ, we decrease the BE throughput and so increase the nrtPS throughput.

In Fig. 11, as we increase the rtPS load, we observe that Strict Priority yields the lowest delay. This is because Strict Priority allocates higher bandwidth for rtPS flows than the others. WFPQ has the highest rtPS delay as its throughput is lower than the others. RED-based WFPQ and Enhanced RED-based WFPQ decrease the delay of rtPS, as they control the weight of rtPS according to the queue length of rtPS.

Fig. 12 shows the variation of nrtPS delay with increasing rtPS load. In this graph, we do not show the results of Strict Priority. The reason is that beyond 1500 kbps the nrtPS flow cannot transmit any packets, and the delay increases extremely. Consequently, the result for SP is not comparable with the other algorithms. The Enhanced RED-based WFPQ algorithm succeeds in decreasing the delay of nrtPS flows over RED-based WFPQ and WFPQ increasing the throughput. Among all three algorithms, WFPQ allocates the highest weight for nrtPS, thus, it has the lowest nrtPS delay.

In Fig. 13, we do not show the results for SP. The reason is that beyond 1500 kbps, the BE flow cannot transmit any packets, and the delay increases extremely. Consequently, the results are not comparable with the other algorithms. Enhanced RED-based WFPQ algorithm increases the delay of BE flows, as the algorithm increases the throughput of nrtPS. Therefore, RED-based WFPQ has lower BE delay than Enhanced RED-based WFPQ. In that point, we provide to transmit BE flows but as BE flow do not have QoS requirement, we increase nrtPS throughput in order to BE's. Among the three algorithms, WFPQ allocates the highest weight to BE, so it yields the lowest delay.

In Fig. 14, we observe that SP yields the lowest dropped packet percentage. WFPQ yields the highest dropped packet percentage, as RED-based WFPQ and Enhanced RED-based WFPQ algorithms increase the throughput of rtPS over that of WFPQ.

According to Fig. 15, WFPQ has the highest number of rtPS packets dropped. As Strict Priority yields the highest throughput for rtPS, its number of dropped packets is the lowest. RED-based WFPQ and Enhanced RED-based WFPQ have the same number of dropped packets. The reason is that their granting mechanism for rtPS flows is the same.

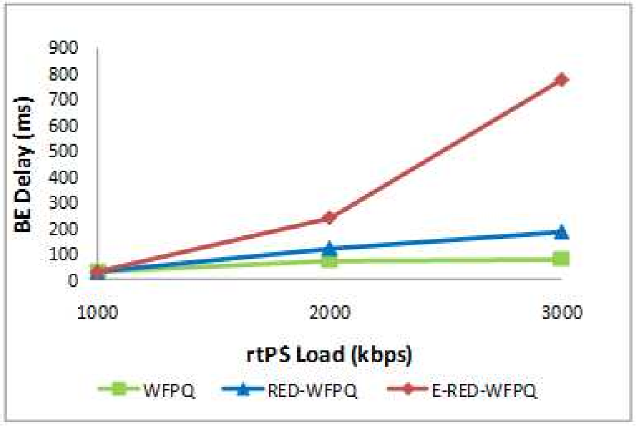

Figure 13. Comparison of BE Delay

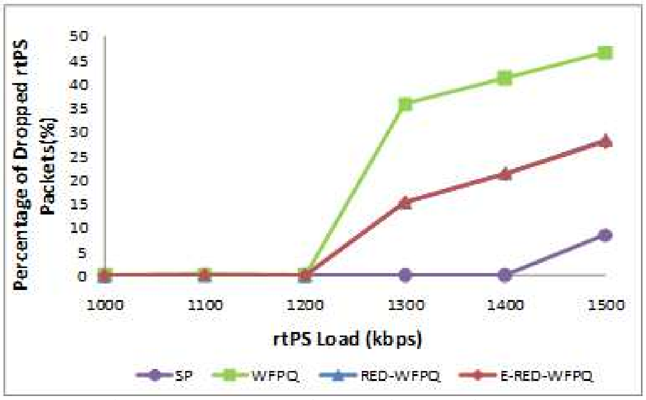

Figure 14. Percentage of Dropped rtPS Packets (%)

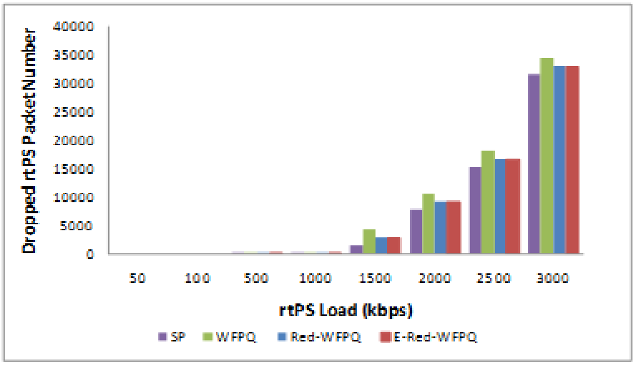

Figure 15. Dropped rtPS Packet Number

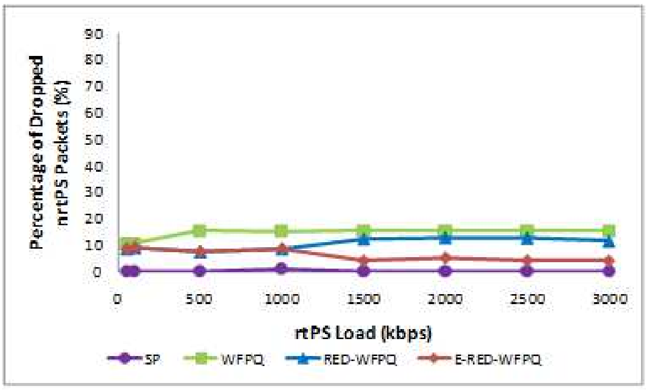

Figure 16. Percentage of Dropped nrtPS Packets (%)

According to Fig. 16, Strict Priority sends nrtPS packets until 1500 kbps rtPS load is reached. Beyond that, due to TCP congestion, nrtPS flows reduce their transmission rate to zero. Since there are no nrtPS packets submitted, the percentage of dropped nrtPS packets equals zero. WFPQ has the highest percentage of dropped nrtPS packets, as RED-based WFPQ and Enhanced RED-based WFPQ increase the nrtPS throughput over that of WFPQ. Also, beyond 1000 kbps rtPS load, Enhanced RED-based WFPQ has a lower percentage of dropped nrtPS packets than RED-based WFPQ. This is to be expected, as Enhanced RED-based WFPQ increases the nrtPS throughput.

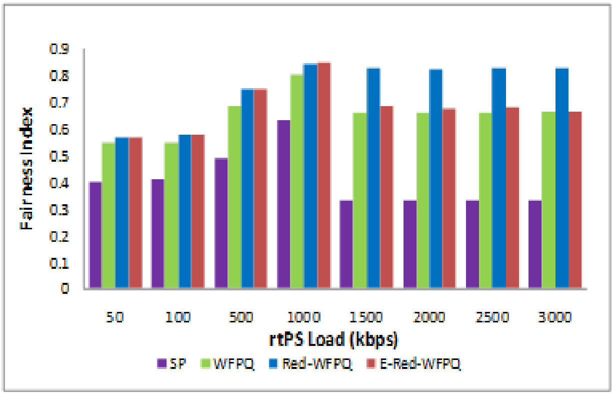

Figure 17. Fairness Index

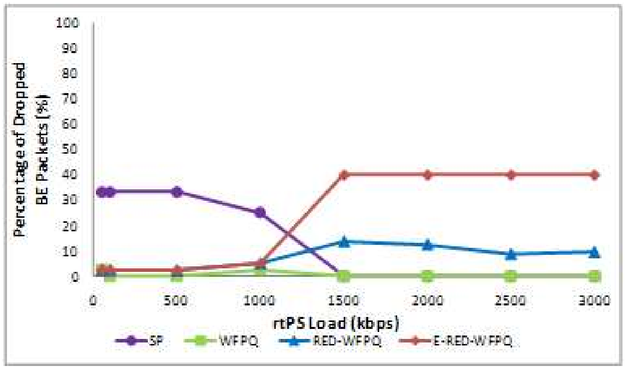

Figure 18. Percentage of Dropped BE Packets (%)

According to Fig. 17, Strict Priority scheduler sends BE packets until 1500 kbps rtPS load is reached. Beyond that, due to TCP congestion, BE flows can not be allocated any bandwidth. Since there are no BE packets submitted, the percentage of dropped packets equals zero. WFPQ has the lowest percentage of dropped BE packets, as RED-based WFPQ and Enhanced RED-based WFPQ decrease the BE throughput. Also, beyond 1000 kbps rtPS load, Enhanced RED-based WFPQ has a higher percentage of dropped BE packets than RED-based WFPQ. This is to be expected, as Enhanced RED-based WFPQ decreases the BE throughput.

Fairness Index is calculated according to [11]. Therefore, we normalized rtPS flow with 64 kbps, nrtPS flow with 45 kbps, and BE flow with 1 kbps. According to Fig. 18, Strict Priority scheduling has the lowest Fairness Index, as it is an unfair algorithm. Enhanced RED-based WFPQ is slightly fairer than WFPQ. As Enhanced RED-based WFPQ allocates sufficient bandwidth for rtPS flows, it achieves higher fairness over WFPQ. RED-based WFPQ has the highest Fairness Index because the algorithm provides more allocation for BE flows. Consequently, normalized values of rtPS, nrtPS, and BE are closer to each other, resulting in a higher Fairness Index. In a QoS-based system, we do not need to provide strong fairness if we increase the QoS of the system. But, we still evaluate if fairness is provided at an acceptable level.

V. CONCLUSION

In this paper, an Enhanced RED-based WFPQ algorithm for SS uplink scheduler is proposed and the throughput of nrtPS is increased while keeping rtPS throughput at high levels. The algorithm is compared with Strict Priority, WFPQ, and RED-Based WFPQ with the respect to throughput, delay, packet loss rate, and fairness. It is observed that the proposed algorithm gives promising results while keeping fairness at reasonable levels among the different QoS classes. The details of the work are available in [12].

Simulation results showed that RED-based WFPQ and Enhanced RED-based WFPQ increase the rtPS throughput, and they follow the same approach while allocating rtPS bandwidth. The rtPS throughput of Strict Priority is the highest and the throughput of WFPQ is the lowest. The nrtPS throughput of Enhanced RED-based WFPQ is higher than that of RED-based WFPQ. The BE throughput of Enhanced RED-based WFPQ is lower than that of RED-based WFPQ. Enhanced RED-based WFPQ increases nrtPS throughput, but it decreases the throughput of BE flows. However, the starvation of BE flows in congested network is prevented.

RED-based WFPQ and Enhanced RED-based WFPQ significantly decrease the delay of rtPS. Strict Priority has the lowest delay, and WFPQ has the highest delay. The delays experienced depend on the throughput of the flows. The nrtPS delay of Enhanced RED-based WFPQ algorithm is lower than that of RED-based WFPQ. The BE delay of RED-based WFPQ algorithm is lower than that of Enhanced RED-based WFPQ.

The number of dropped rtPS packets is directly proportional to the delay; therefore, Strict Priority exhibits the lowest number of dropped rtPS packets, while WFPQ exhibits the highest number of dropped rtPS packets. The number of dropped rtPS packets for RED-based WFPQ and Enhanced RED-based WFPQ are the same.

Performance of the studied and proposed schedulers is given in terms of Fairness Index also. It is observed that, SP scheduling is unfair and its Fairness Index is the lowest. RED-based WFPQ and Enhanced RED-based WFPQ have higher Fairness Index than WFPQ. The reason is the allocation of the bandwidth depend on the queue length and shows dynamic characteristic.

Currently we are working on dynamically changing weight thresholds. Here, the thresholds could adapt to the state of the service flow queues.

ACKNOWLEDGMENT

This project is partially supported by Istanbul Chamber of Industry and TUBITAK.

REFERENCES

[1] Borin, J. F. and Fonseca, N. L. S. da, “Simulator for WiMAX networks”, Elsevier Simulation Modeling Practice and Theory 16, 1, pp. 817-833, 2008.

[2] IEEE Standard 802.16-2004, “IEEE standard for Local Metropolitan Area Networks, Part16: Air interface For Fixed Broadband Wireless Access Systems ”, June 2004.

[3] IEEE Standard 802.16e-2005, “IEEE standard for Local Metropolitan Area Networks, Part16: Air Interface for Fixed Broadband Wireless Access Systems - Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands”, December 2005.

[4] Pratik, D. et al., “A Performance Study of Uplink Scheduling Algorithms in Point-to-Multipoint WiMAX Networks”, Elsevier Computer Communications 32 511-521, 2009.

[5] Masri et al., “An Uplink Bandwidth Management Framework for IEEE 802.16 with QoS Guarantees”, IFIP Networking, Aachen, Germany, 2009.

[6] Po-Chun Ting, Chia-Yu Yu, and Naveen Chilamkurti., “A Proposed RED-Based Scheduling Scheme for QoS in WiMAX Networks”, IEEE International Symposium on Wireless Pervasive Computing (ISWPC), February 2009.

[7] <http://w3.antd.nist.gov/seamlessandsecure/>, last accessed in May 2010.

[8] <http://www.isi.edu/nsnam/ns/>, last accessed in July 2010.

[9] NIST, Seamless and secure mobility, NS-2 NIST WIMAX Module, http://www.antd.nist.gov/seamlessandsecure, May 2010.

[10] A. Belghith and L. Nuaymi, "Design and Implementation of a QoS-included WiMAX Module for NS-2 Simulator", Simutools, ACM, 2008

[11] Pratik, D., 2007. A Performance Study of Uplink Scheduling Algorithms in Point to Multipoint WiMAX Networks: MSc Thesis, *Queen’s University, Canada*.

[12] Kasaci, S., 2010, An Enhanced RED-Based Weighted Fair Priority Queuing Algorithm for IEEE802.16 SS Scheduler, MS Thesis, Istanbul Technical University, Institute of Science and Technology.

ISBN: 978-1-61208-103-8