# A New Queue Length based Scheduling Strategy for nrtPS Service Class in IEEE 802.16 Networks

Pallavi Grover[1] and Marcio Andrey Teixeira[2]
[1]M.E Research Scholar, Comp Sc. & Engg., UIET Panjab University, Chandigarh, India
[2]Department of Computing, Federal Institute of Education, Science and Technology of São Paulo, Catanduva, Brazil

Keywords: IEEE 802.16 Networks, nrtPS, QoS, Scheduling Algorithms.

Abstract: IEEE 802.16 standard is designed to provide services to various types of multimedia applications. It supports real-time and non-real-time service classes. With such a large volume of traffic, a new strategy needed to be developed for the non-real-time service class, since there have been limited studies in this area. In this paper, a new queue length based scheduling strategy for the non-real-time service class is proposed. The proposed algorithm is developed on the basis of virtual queue's and counter scheme's, aiming at ensuring minimum bandwidth for non-real-time applications.

## 1 INTRODUCTION

In recent years, the development of wireless technology has increased rapidly, particularly in the field of broadband wireless networks. WiMAX (IEEE, 2013) is intended to offer low-cost, high-speed internet access to a wide variety of devices. The advantages of WiMAX networks include high transmission speed, scalable bandwidth, link layer retransmission, robust security, high peak data rates, and an efficient QoS mechanism for data, voice, and video. It is very challenging task to assure QoS requirements of every type of traffic in a wireless network. The challenges associated with WiMAX (IEEE, 2013) are due to limitations of wireless networks such as strong attenuation with increasing distance, Rayleigh fading, limited scalability etc. The protocol stack of IEEE 802.16 standard which defines characteristics of physical layer's as well as media access control (MAC) layer's.

The working architecture of the IEEE 802.16 network is composed of 2 components: Base Station (BS) and Subscriber Station (SS). The BS is a core element that acts as an interface between the infrastructure network such as the Internet Service Provider (ISP) and the SS. The SS further extends internet services to its users. The SSs connects with the BS to provide a variety of services to its users. WiMAX has defined five service classes: Unsolicited Grant Service (UGS), real-time Polling Service (rtPS), non-real-time Polling Service (nrtPS), best-effort service (BE) and an extended real-time Polling Service (ertPS). Each of them is associated with a certain set of QoS parameters such as delay, throughput, and jitter.

Real-time services such as UGS, ertPS, and rtPS are always given higher priority by the scheduler, whereas non-real-time traffic, which amounts to the majority of internet traffic, is always neglected. With such large amounts of traffic volume, there is a need to develop a dedicated strategy for the non-real-time service class. There have been very limited studies in this regard.

The IEEE 802.16 standard does not specify any scheduling algorithm, it is left to the vendors as to whether or not they will employ their own scheduling algorithm. Scheduler defines the distribution of allocated resources in the form of slots, which are further mapped into sub channels. Logically, it calculates the number of slots for each class and physically, it selects the sub channels and time intervals. There are 3 distinct scheduling processes: two are for the BS (uplink and downlink), and one is for the SS (uplink). The proposed algorithm is developed to act in the uplink scheduling at BS, because the uplink scheduling is more complex than the downlink scheduling. In the downlink scheduler, the BS has complete knowledge of the queue status whereas in the uplink scheduling, the input queues are located in the SSs and hence, are separated from the BS. The BS does not have any information about the

Grover, P. and Teixeira, M.
A New Queue Length based Scheduling Strategy for nrtPS Service Class in IEEE 802.16 Networks.
DOI: 10.5220/0005989601650171
In *Proceedings of the 13th International Joint Conference on e-Business and Telecommunications (ICETE 2016) - Volume 6: WINSYS*, pages 165-171
ISBN: 978-989-758-196-0

arrival time of packets in the SS queues, however, the BS makes the bandwidth allocation based on the bandwidth requests (BW-REQ), messages sent by the SSs. The BW-REQ message indicates the queue status in the SS.

This paper presents a survey of scheduling algorithm to the nrtPS connections and proposes a new scheduling algorithm for nrtPS service class based on virtual queue and counter scheme. The proposed algorithm is applied directly to the uplink virtual queue in the BS aiming at ensuring minimum bandwidth for nrtPS service class. The scheduling is done by assigning priority to the connection having larger queue size. Counter is attached to each virtual queue to prevent starvation of connections having lower queue size. The proposed algorithm has been evaluated by means of modeling and simulation. The simulations experiments have shown satisfactory results.

The remainder of this paper is organized as follows: Section II resumes the related work. Section III describes the proposed work. Section IV defines the network scenario and the main parameters used in the simulation. Section V shows the numerical results. Finally, Section VI concludes the paper.

## 2 RELATED WORK

The literature survey of non-real-time services can be traced back to conventional algorithms such as proportional fairness and modified largest weighted delay first algorithms. Proportional fairness was originally designed for downlink traffic to increase the throughput of the system, as well as to provide fairness among multiple queues. It calculates a priority function which is the ratio of the current rate, to the average rate, and then schedules different queues accordingly. It is very simple and efficient, but it does not take into consideration saturated queues, when dealing with non-real-time traffic.

Qingwen Liu et al. (Liu, 2006) proposed a cross layer scheduling algorithm providing QoS support in IEEE 802.16 networks. A priority is assigned to each connection which is updated dynamically on basis of wireless channel quality and QoS satisfaction. For each nrtPS connection, a minimum reserved rate (ŋ) is defined. The proposed algorithm ensures average transmission rate should be greater than ŋ. Scheduling of nrtPS connections is done with the help of priority function, which is dependent on nrtPS-class coefficient and rate satisfaction indicators. The rate satisfaction indicator is the ratio of the average transmission rate over the minimum reserved rate. If the value of the indicator is greater than 1, then the requirement is satisfied; otherwise, packets should be sent as soon as possible. This scheduler offers flexibility, scalability, and low implementation complexity. The major drawbacks include fairness issues among same service classes and imperfect channel conditions arising, due to errors in estimation and feedback latency.

Authors Fen Hou et al. (Hou, 2009) presented a simple scheduling structure for non-real-time services in the IEEE 802.16 networks. It is a cross layer algorithm, which considers selective automatic repeat request mechanisms at MAC which layers and uses adaptive modulation and coding schemes at the physical layer. It tries to ensure minimum throughput requirements of the nrtPS class, and at the same time, maintains flexibility between resource allocation and packet scheduling. To achieve this flexibility, two parameters, m and n are defined, where m represents the number of SSs selected in each MAC frame, and n represents bandwidth, which is granted to SS when it is being serviced. In the beginning of each frame m, SSs, which have superior channel conditions, are selected and n, amounts to the number of resources that are given to them. The value of m=1 assures opportunistic scheduling, leading to maximum resource utilization. When m = the total number of SSs, then it leads to minimum resource utilization with lesser delivery delays. This paper focuses on scheduling unicast nrtPS applications, but there is a dire need to concentrate on multicast multimedia applications.

Ali Mohammed Alsahag et al. (Alsahag, 2014) proposed a fuzzy based adaptive deficit round robin uplink scheduler that adjusts the weights of the service queues for real-time as well as non-real-time applications. The allocation of bandwidth is done on the basis of deadline based schemes. To compute the deadline authors have used maximum latency for real-time and throughput for non-real-time as input variables. The overall mechanism can be divided in three fundamental phases: fuzzification, fuzzy inference, and defuzzification. In the fuzzification process, we use two input variables as real-time maximum latency (RTML) and non-real-time throughput (NRTTHR). These input variables are processed with the help of a rule base in the fuzzy inference phase, and then finally in the defuzzification phase, where crisp numerical values are obtained, which determines the weights that need to be used, as an indication for priority. In the bandwidth assignment process, several queues are maintained which are associated with a DC value. In each round DC is incremented by a value, which is determined by the fuzzy system, keeping in consideration overall capacity of the system. Transmission of queue takes place when the DC value is equal to the amount of requested bandwidth. The proposed algorithm optimizes the overall systems utilization, but gives more consideration to satisfying maximum latency requirements of real-time traffic.

D. David Neels Pon Kumar et al. (Kumar, 2011) proposed a neural network based on fuzzy priority scheduling algorithms. Fuzzy is used to calculate priority, which itself is composed of a primary fuzzy scheduler and a dynamic fuzzy scheduler. The primary fuzzy scheduler takes an input Expiry time (E), Waiting time (W), Queue length (Q), Packet size (P) and gives priority index as output. After this priority index is feed as an input to the dynamic fuzzy scheduler, which calculates the final priority, by taking certain types of services into consideration. Scheduling is done using an artificial neural network, based on prioritized input received from DPFS (dynamic priority fuzzy scheduler).Artificial neural network (ANN) consists of three layers: input layer, modified Kohonen layer and the Grossberg layer. The input layer processes prioritized output received from DPFS and arranges them in order of priority. After, the output is fed to modify the kohonen layer, which checks whether the value is in the range of the threshold. If it is given as an input to the grossberg layer it has less chance to be rejected. This algorithm improves fairness, and prevents starvation of the low priority traffic, but it has not considered bursty traffic conditions. The major drawback is that processing time has increased.

Dusit Niyato et al. (Niyato, 2007) presented a survey of the game theory techniques for management of radio resources in different wireless networks and also proposed bandwidth allocation and connection admission control schemes for IEEE 802.16 networks. The bandwidth allocation helps the non-cooperative game in which three parameters are considered: players, strategies, and pay off's. Players refer to rtPS and nrtPS connection, strategies which imply the amount of bandwidth that is to be allocated with a new connection and payoff's that refer to the total utility of currently running rtPS and nrtPS connections, plus the utility of new connections. The solution of this game is provided with the help of nash equilibrium which is calculated through best response function. It maximizes the payoff of BS which defines QoS requirements of connections. The simulation results of adaptive scheme shows that it is not able to satisfy delay and throughput necessities when load is high.

# 3 PROPOSED SCHEDULING ALGORITHM

The proposed scheduling algorithm aims at guaranteeing the efficient utilization of the bandwidth resources, and thus, promotes the effective use of the wireless link. Figure 1 shows the architecture used to develop the proposed solution.

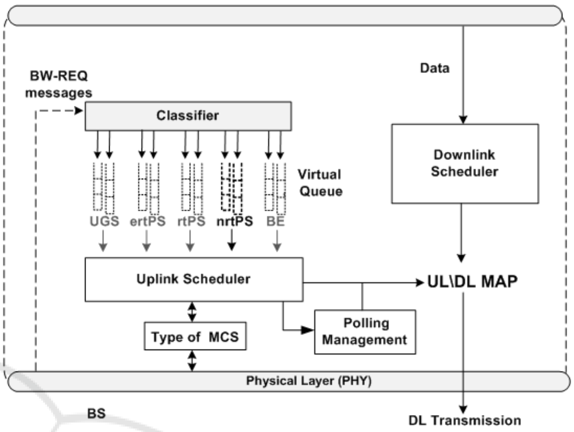

Figure 1: Architecture used in development of the proposed solution.

As shown in Figure 1, the BS receives from the SS bandwidth request (BW-REQ) message, which reports the current queue size of each connection. The algorithm is applied directly to the nrtPS bandwidth request which queues in the BS (Teixeira, 2012). Figure 2 shows the pseud-code of the proposed scheduling algorithm.

The scheduling process of the nrtPS is explained below, in the following steps:

1) Initially, the algorithm checks the amount of bandwidth requested by the nrtPS connections, and stores them in a virtual queue at the BS (lines 1 - 7).

2) It then verifies the actual amount of bandwidth requested by the SSs and sorts the bandwidth requests by the largest queue size (lines 8 - 15).

3) After sorting the bandwidth requests, a counter is assigned to each virtual queue, which prevents the starvation of connections, which ultimately prevents having lower bandwidth requests (line 16).

4) On completion of the first round of allocation, the proposed algorithm verifies if the bandwidth requests are satisfied or not. If it is not satisfied, the algorithm checks if there are more symbols to allocate (lines 18 - 25). However, in this case, the scheduling sorts the connections by the highest counter number.

5) If more symbols are available, then the proposed algorithm allocates these available symbols

to nrtPS connections, and its counter value will be decreased (line 29). When the counter decreases to zero, the count is initiated (lines 26-27).

```
1: Verifies the bandwidth messages request at the BS queue;
2: begin
3:  for BWrequest i at the BS queue do
4:   begin
5:     if (BWrequest[i] = nrtPS) then
6:       begin
7:         BW[i] += BWrequest[i].length;
8:         if (connections_numbers > 1) then
9:           begin
10:            if ( BWrequest[i].length < BWrequest[i+1].length) then
11:              begin
12:                Temp = BWrequest[i].length;
13:                BWrequest[i].length = BWrequest[i+1].length;
14:                BWrequest[i+1].length = Temp;
15:              end;
16:                BWrequest[i].counter = 3;
17:              UL-MAP = request order by the length;
18:            if (BwToAllocate > 0) then
19:              begin
20:                if ( BWrequest[i].counter < BWrequest[i+1].counter) then
21:                  begin
22:                    Temp = BWrequest[i].counter;
23:                    BWrequest[i].counter = BWrequest[i+1].counter;
24:                    BWrequest[i+1].counter = Temp;
25:                  end;
26:                  if (BWrequest[i].counter == 0) then
27:                      BWrequest[i].counter =3
28:                  else
29:                      BWrequest[i].counter --;
30:                UL-MAP = request order by the counter;
31:              end;
32:          end;
33:     end;
34:  end;
35:end;
```

Figure 2: Proposed Scheduling Algorithm.

Figure 3 shows the Flowchart of the proposed scheduling algorithm.

## 4 MODELING AND SIMULATION

The simulation studies and evaluates the properties of the proposed scheduling algorithm for non-real-time traffic. It has been implemented in the Network Simulator 2 (NS-2) version 2.34 (Network, 2016) along with the WiMAX module. The simulation scenario consists of a BS, and also of several SSs distributed around the BS in a random mode. The Table 1 shows the main parameters used in the simulation.

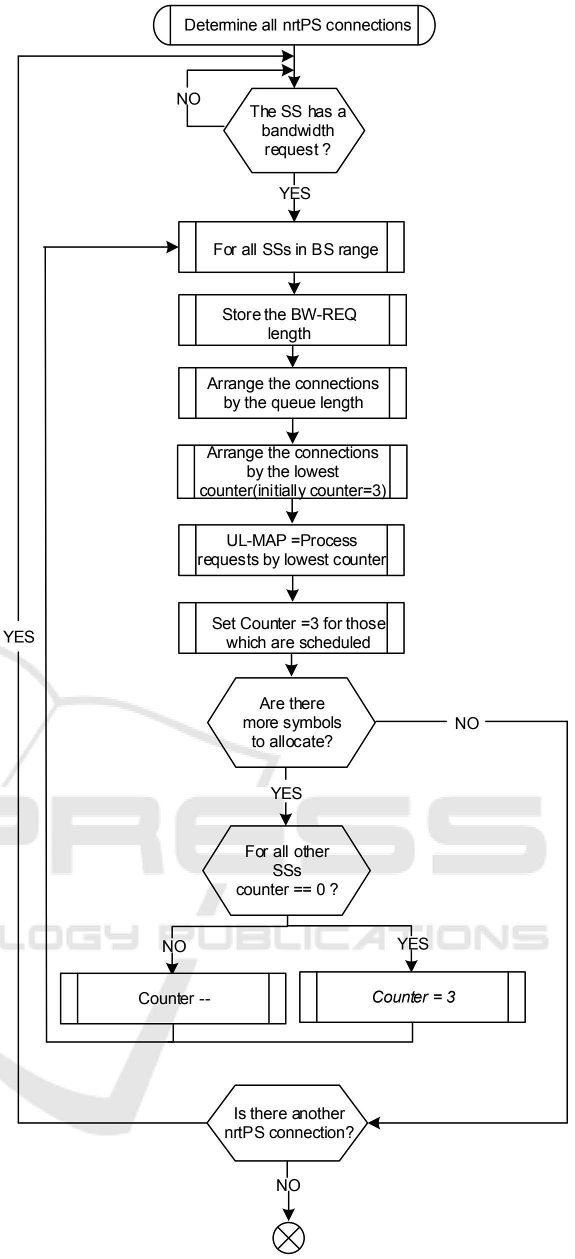

Figure 3: Flowchart of the proposed scheduling algorithm.

Table 1: Main parameters used in the simulation.

| Parameters | Values |
|---|---|
| Frequency band (MHz) | 5 |
| Transmit antenna gain | 1 |
| Received antenna gain | 1 |
| Frame duration (ms) | 20 |
| Cyclic prefix | 0.25 |
| Simulation time (s) | 100 |
| Downlink bandwidth (Mbps) | 5.4 |
| Uplink bandwidth (Mbps) | 10.5 |

The sources of traffic used in the simulation were, voice, video, web and a file transfer, which corresponds to the UGS, rtPS, BE, and nrtPS class. The voice traffic is modeled as an on/off scheme with a mean of 1.2 s and 1.8 s for "on" and "off" periods respectively. During "on" period packets, 66 bytes are generated every 20 ms, which follows an exponential distribution. The voice traffic was modeled on a basis of an exponential traffic model in NS-2.

The video was modeled using a traffic source that generates packets periodically with variable sizes, simulating the MPEG traffic. The packets size vary between 450 to 1500 bytes. The web traffic was modeled using hybrid Lognormal/Pareto distribution. FTP traffic was generated from a source which follows exponential distribution within a mean of 512 Kbytes.

The performance of the proposed algorithm was evaluated to take into account the follow performance metrics:

**Average end-to-end-delay**: This is the time taken by packets to transmit from source to destination. There can be various reasons for the delay such as propagation and network delay, source and destination processing delay etc. It can be calculated as followed:

$$Average\ Delay = \frac{\sum(Packet\ arrival\ time - Packet\ start\ time)}{\sum(Number\ of\ Connections)} \quad (1)$$

**Throughput**: This measures the data rate which is generated by the application in terms of bits per second. It can be calculated as followed:

$$Throughput = \frac{\sum Packet\ Size}{\sum(Packet\ arrival\ time - Packet\ start\ time)} \quad (2)$$

The nrtPS scheduler should guarantee minimal bandwidth to the users. It is used with the Throughput metric. However, the average delay performance metric is used to analyze whether the proposed scheduler algorithm is influencing the other service classes (UGS, rtPS BE).

# 5 NUMERICAL RESULTS

## 5.1 Experiment 1

The first experiment analyzes the performance of the nrtPS connections and verifies how the BS allocates bandwidth to the nrtPS service class. The simulated network consists of one BS and a number of nrtPS SSs, varying between 2 and 12. The transmission rate of each nrtPS connection is 512 kbps. There are 2 UGS, 4 rtPS, and 8 BE connections as background traffic, along with nrtPS connections. Each UGS, rtPS and BE connection generates 40, 320 and 510 kbps respectively. The Figure 4 shows the behavior of average delays of nrtPS SSs.

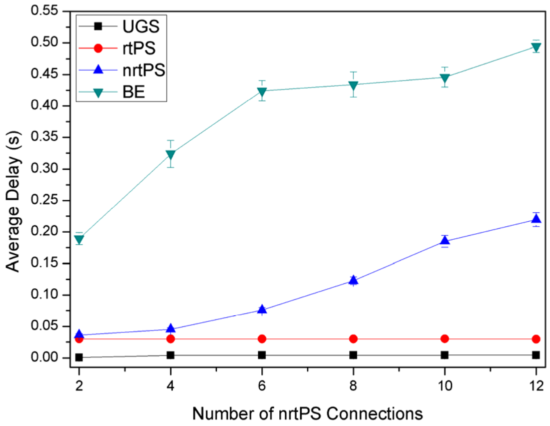

Figure 4: Average delay of the UGS, rtPS, nrtPS and BE connections.

It is possible to see in Figure 4 that the average delay rises with an increase in the number of nrtPS connections. The increase in the average delay has occurred due to link saturation and the total traffic of the nrtPS connections has increased. The proposed algorithm prioritizes transmission of nrtPS connections on basis of queue length, which results in lesser average delay. The average delay of the UGS and rtPS classes remains constant, as defined by the standard. The increased of the nrtPS load traffic does not interfere in the UGS and nrtPS classes. This is expected because the UGS and rtPS Traffic have high priority over nrtPS and BE service classes. The average delay of the BE is high, however, the BE service class does not have QoS parameters.

Figure 5 shows the throughput of the UGS, rtPS, nrtPS and BE connections.

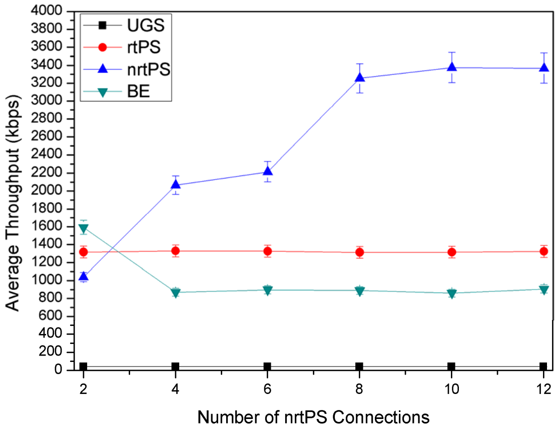

Figure 5: Throughput of the UGS, rtPS, nrtPS and BE connections.

Figure 5 shows the increase of the nrtPS traffic load also does not interfere with the UGS and rtPS traffic. The BS distributed the resources to all service classes. However, with the increase of the nrtPS traffic load, the throughput of the BE connections decrease. The BE connections have lower priority among the other service classes, which causes them to receive less resource (slots).

## 5.2 Experiment 2

The second experiment compares the performance of the proposed algorithm with the traditional Round Robin (RR) scheduling algorithm. This simulated network consists of one BS a number of nrtPS connections varying from 2 to 12. There are 2 UGS, 2 rtPS and 4 BE connections as background traffic along with nrtPS connections. Figure 6 shows the throughput of the RR and the proposed algorithm.

As we can see in Figure 6, the proposed algorithm has better performance than the RR scheduling algorithm. This happens because the proposed algorithm organizes the uplink frame in accordance with the queue length. Moreover, the algorithm verifies the counter of the nrtPS connections to avoid starvation. The RR scheduler makes the scheduling and does not consider the queue length.

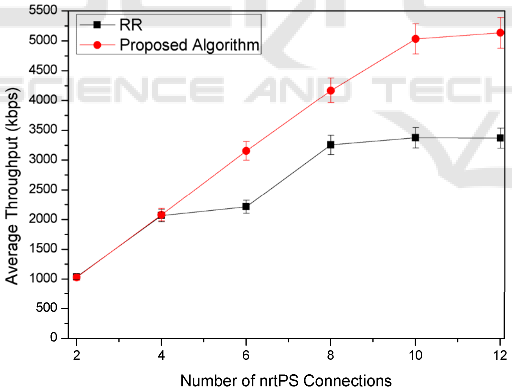

Figure 6: Throughput of the RR and nrtPS connections.

## 5.3 Experiment 3

The third experiment verifies the performance of the nrtPS connections in the presence of high and low priority traffic. The input traffic has increased by a ratio of 2:2:2:2 i.e. 2 UGS, 2 rtPS, 2 nrtPS and with 2 BE connections. In such a scenario, performance metrics are analyzed by increasing traffic inputs up to 10 connections of each service class. This experiment helps in analyzing the performance of the scheduler in presence of diverse traffic classes. Figure 7 shows the throughput of the connections.

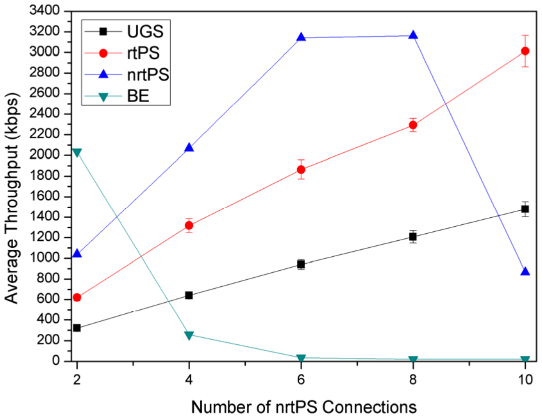

Figure 7: Throughput of the UGS, rtPS, nrtPS and BE connections.

It is possible to see in Figure 7 that the throughput of the nrtPS connections decreases with the increase of the traffic load in the system. However, the scheduler distributes the bandwidth to guarantee minimum bandwidth to the nrtPS connections. This decrease in throughput can be justified due to the presence of high priority traffic, along with nrtPS connections. Increasing demand for these classes forces the system to allocate more amounts of bandwidth to them instead of the nrtPS connections. However, the proposed algorithm allocates bandwidth efficiently to the nrtPS connections in presence of UGS and rtPS connections, which are highest in priority. When the number of connections increases, the scheduler distributes the resources to connections with high priority and the BE connections receives fewer resources once the BE connections do not have any QoS requirements.

The same experiment has been completed, but this time, with the RR algorithm, in order to compare the performance between RR and the proposed algorithm. The Figure 8 depicts the throughput of the proposed algorithm and the RR.

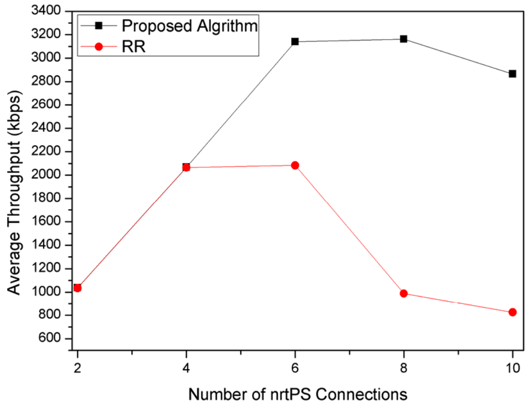

Figure 8: Throughput of the RR and nrtPS connections.

As is shown in Figure 8, the proposed algorithm allocates the bandwidth more efficiently than the RR algorithm. The resources were distributed more efficiently when the scheduler was made using the information about the queue length. Furthermore, the counter scheme helps the scheduler to avoid the starvation to the nrtPS connections that have lower queue length.

## 6 CONCLUSIONS

In this paper, a queue length based scheduling strategy for an nrtPS service was proposed. The proposed algorithm was verified by performing different experiments with diverse traffic scenarios. The performance metrics used to evaluate this proposed scheduling algorithm were average delay and throughput. Scheduler is expected to allocate bandwidth effectively to the nrtPS class. It should be able to satisfy the constraint of minimum throughput for the nrtPS class, even in the presence of high priority traffic classes such as UGS and rtPS. It has been observed that bandwidth allocation to nrtPS class is done efficiently because it leads to increased quality and more user satisfaction. The proposed algorithm also shows better performance than the RR scheduling algorithm.

## REFERENCES

IEEE Std. 802.16n 2013. IEEE standard for local and metropolitan area networks Part 16: Air interface for fixed and mobile broadband wireless access systems—amendment 2: Higher Reliability Networks. *IEEE Computer Society and IEEE Microwave Theory and Techniques Society*. p. 1–132. ISBN 978-0-7381-8425-8, 2013.

Liu Q, Wang X, Giannakis G.B., 2006. A Cross-Layer Scheduling Algorithm With QoS Support in Wireless Networks, *IEEE Transactions on vehicular technology*; 55(3), 839 – 847.

Hou F, Ho P.H.,2009. A Flexible Resource Allocation and Scheduling Framework for Non-real-time Polling Service in IEEE 802.16 Networks, *IEEE transactions on Wireless Communications*; 8(2): 766 – 775.

Alsahag A.M, Ali B.M, Noordin N.K, Mohamad H., 2014. Fair uplink bandwidth allocation and latency guarantee for mobile WiMAX using fuzzy adaptive deficit round robin, *Journal of network and computer applications*;39:17-25.

Kumar D.D.N.P, Murugesan K, Raghavan S, Suganthi M., 2011. Neural Network based Scheduling Algorithm for WiMAX with improved QoS Constraints In: *Proceedings of international conference on Emerging trends in electrical and computer technology, 23-24 March, Tamil Nadu*. p. 1076 – 1081.

Niyato D, Hossain E., 2007. Radio resource management games in wireless networks: an approach to bandwidth allocation and admission control for polling service in IEEE 802.16, *IEEE Journal of Wireless Communications*;14(1):27–35, Feb.

Teixeira, M. A.; Guardieiro, P. R., 2012. Adaptive packet scheduling for the uplink traffic in IEEE 802.16e networks. *International Journal of Communication Systems (Print)*.

The Network Simulator-ns2.