

Ageing and Priority-Based Scheduling for Uplink in WiMAX Networks



Deepa Naik, Himansu Rathi, Asish Dhara and Tanmay De

Abstract WiMAX is an emerging technology for next-generation wireless networks which supports a large number of users in an economic way. To achieve quality of service (QoS) requirements, an efficient and reliable scheduling algorithm is needed. The existing approaches in the literature have been proven to provide the best performance in allocating bandwidth to subscriber stations to maximize the throughput and ensure the constraints of delay in each class of traffic. In these approaches, starvation of lower priority class was not considered, which is of great significance in reality. In this paper, we have considered the starvation of lower priority classes to provide QoS requirement to each class of traffic in an acceptable way, and a bandwidth scheduling algorithm is proposed. A comparative study between the proposed scheduling algorithm and the existing scheduling algorithm shows the better performance in terms of maximizing the network throughput in given network, while minimizing the starvation of lower priority classes.

Keywords WiMAX (world interoperability for microwave access network) · QoS (quality of service) · UGS (unsolicited grant service) · RTPS (real-time polling service) · nRTPS (non-real-time polling service) · BE (best effort)

D. Naik (✉) · H. Rathi · A. Dhara · T. De
Department of Computer Science and Engineering,
National Institute of Technology, Durgapur, India
e-mail: naiksavantdeepa@gmail.com

H. Rathi
e-mail: himansurathi@gmail.com

A. Dhara
e-mail: ashishere15@gmail.com

T. De
e-mail: tanmayd12@gmail.com

1 Introduction

Fixed WiMAX based on IEEE 802.16 standard is cost effective for fixed wireless networks. IEEE 802.16e is the new version to the fixed WiMAX mobility. WiMAX provides high data rate mobile wireless services for metropolitan areas. WiMAX coverage range is up to thirty-mile radius and data rates between 1.5 to 75 Mbps theoretically. The WiMAX defines the two modes of operation namely mesh and point-to-multipoint (PMP) mode. In mesh mode, each subscriber station (SS) can communicate to each other and to the base station (BS). In point-to-multipoint mode, subscriber stations can communicate only through base station. The base station is responsible for providing the QoS to each of its subscriber stations.

The scheduling algorithm must be simple to implement because real-time application supported by subscriber station needs the quick response from the centralized base station. So the time complexity of scheduling algorithm must be simple. Lots of research on scheduling algorithm has been investigated in [1, 2]. The issues in allocating resource are more challenging. In our proposed scheduling algorithm, base station uses the ageing and priority-based scheduling (APBS) to schedule the traffic of different classes. The downlink scheduler at the base station schedules the entire bandwidth among the subscriber stations depending on the grant per connection type.

The rest of the paper is organized as follows: Sect. 2 presents the previous works. In Sect. 3, problem is defined in formal notation. The proposed approach is presented in Sect. 4. The complexity analysis and results are described in Sect. 5 and Sect. 6, respectively. Finally, conclusion and future work are drawn in Sect. 7.

2 Previous Works

Our study focuses on centralized scheduling in point-to-multipoint (PMP) mode. In the literature, many scheduling algorithms have been proposed for downlink scheduling. There are different types of scheduling algorithm, namely traditional, dynamic scheduling. In [3] traditional scheduling algorithms uses the same Schelling technique as in computer operating system. It shares the equal network resources to all the subscribers without concern about the priority. However, this technique is not suitable for WiMAX scheduling, where the traffic demand of varying size arrives in random fashion. The authors Sagar et al. in [4] proposed the weight round robin (WRR) for scheduling the bandwidth to each queue. But it is not suitable for WiMAX scheduling because the main drawback of WRR is that when the traffic has a variable packet size, WRR provides an incorrect percentage of bandwidth allocation.

The dynamic schedulers are adaptive in bandwidth allocation. The authors Fathi et al. [5] proposed joint scheduling and call admission control (CAC) technique for scheduling packets. In [6] proposed the joint routing and scheduling algorithm for

WiMAX network with the provision for fairness in bandwidth allocation and increase in throughput.

Best of our knowledge, no work has been conducted to overcome the starvation of lower ordered traffic request, while maintaining the quality of service to each class of traffic.

3 Problem Formulation

The WiMAX network consists of base station (BS), subscriber station (ss) and users. The base station is connected to subscriber station in point-to-point mode. The user under the coverage of base station is directly connected through wireless links. The user not under the coverage area of base station is connected through subscriber stations. Given a network topology, which is a directed graph $G(V,E)$, where V is a set of subscriber stations/user and E is the set of bidirectional wireless link between the base station and subscriber station/user at the particular instant of time. 'C' is the capacity of each link between the base station to subscriber station/users. The base station is responsible for allocating network resource to subscriber stations. The connection requests from all class of traffic are represented by notation $(S, B, requested, Class_{type}, B, start_{timeofrequest}, request.duration)$, where $S, B, requested$ and $start_{timeofrequest}, request.duration$ represent the source node, bandwidth demanded and arrival time of each request and duration of the request, respectively. The centralized uplink bandwidth scheduling with the prime objective is to maximize the network throughput and meanwhile eliminate the starvation due to the higher class traffic is considered.

Assumptions:

We use the following assumptions in this work.

- Traffic demands is static in nature. i.e. traffic demands are known in advance.
- The total number of traffic is $R_i = N$.
- w_{Mn} is 1 if the relay station M associated to base station n , $M \in ss$ and $j \in BS$ 0 otherwise.
- R_i , where M_i^{ss} is the number of subscriber station(M)/user including the traffic class i in the network.
- $B_{k,i}$ bandwidth request of k th subscriber/user with traffic class belong to i th class of traffic.
- β_n is 1 if the BS is installed at site n , $n \in BS$ 0 otherwise.
- γM is 1 if the ss is installed at site M , $M \in ss$ 0 otherwise.
- We have reserved 60% of bandwidth for UGS traffic in terms of bytes per seconds (Bps), and 20% is reserved for RTPS, and 10% is reserved for nRTPS and BE. However these are not rigid limits and are flexible if requests of another class are missing or are lesser.
- α value is assumed as 0.5.

Objective:

The objective here is to maximize the network throughput, while meeting the quality of service to each class of traffic requirement of WiMAX standard. The objective function to maximize the network throughput is defined as follows:

$$R_i = \sum_{k=1}^{M_i^{ss}} B_{k,i} \quad (1)$$

$$\text{Maximize } \sum_i^N R_i * p \quad (2)$$

$$p = \begin{cases} 1, & \text{if } R_i \text{ is established.} \\ 0, & \text{otherwise.} \end{cases}$$

Subjected to:

Base station to relay station constraints

$$\sum_{\forall n \in BS} w_{Mn} = \gamma_M \quad \text{where } \forall M \in SS \quad (3)$$

$$w_{Mn} \leq \beta_n \quad \text{where } \forall M \in SS \quad n \in BS \quad (4)$$

Equation 3 ensures the constraint that each relay station is connected to only one base station. Equation 4 assures the base station to relay station connection. The number of base station is less than or equal to the relay station installed.

$$\sum_{i=1}^n T_{ss_i} \leq T \quad (5)$$

$$\sum_{i=1}^n T_{ss_i} \times \beta \leq T \quad (6)$$

$$\beta = \begin{cases} 1, & \text{if } ss_i \text{ used time slot } T_{ss_i} \text{ and associated with base station BS.} \\ 0, & \text{otherwise.} \end{cases}$$

ss_i refers to the number of subscriber station attached. BS refers to the number of base station. T_{ss_i} refers to the time slot allocated to each subscriber station. Equation 5 indicates the number of time slot allocated to each subscriber station should be less than or equal to the total number of time slot(T) available with base station BS .

4 Proposed Approach

In this section, the proposed algorithm (APBS) is presented with the objective to maximize the throughput of given network. Our algorithm is categorized in two stages. The first stage of our algorithm is to assign the base station to subscriber stations/nodes under the coverage area. Base station executes the uplink scheduler at every frame interval of time (t) and sends the grant requests to subscriber station in uplink map message. In the second stage, each subscriber station associated with four different class of traffic, namely UGS, RTPS, nRTPS and BE are checked for the expire time and stored in the list L_{req} . The base station maintains four types of priority queues. The request is stored in respective priority queues (UGS, RTPS, nRTPS, BE). The scheduler function is called to schedule the requests of each base stations. This type of scheduling is known as grant per connection (GPC). The algorithm, namely Ageing and Priority-Based Scheduling (ABPS), is depicted in Algorithm 1, to schedule the traffic of different classes. The downlink scheduler at the base station schedules the entire bandwidth among the subscriber stations. Scheduling technique considers the two factors, while serving the request. The scheduler has the age and deadline associated with each class of traffic. The scheduler calculates the weight associated with each request in each queue and stores it in the priority queue of respective class. The scheduler schedules each request before the deadline. The early deadline requests are served first. When alpha is 0, then the priority of all the requests is same and only the time in which they arrive is the only measure that we use to sort the requests; hence this turns out to be FCFS scheduling. And for any other value of alpha, we have taken both age and deadline of the requests into consideration, so it turns out to be APBS algorithm.

$$Age = current_{timeofsystem} - start_{timeoftherequest} \quad (7)$$

$$Deadline = End_{timeofrequest} - current_{timeofsystem} \quad (8)$$

$$Weight = \frac{\alpha \times (age + 1)}{(deadline + 1)} \quad (9)$$

α can vary between 0 and 1.

First, the UGS traffic is admitted in priority queue and calculate the weight associated with each request. The UGS connection is served depending on the bandwidth requirement. The request is served in queue depending on the deadline. The request with early deadline is served first. In this case, $\sum_{i=1}^n B_{UGS}$ are the total request with early deadline in UGS class, are served first. The remaining request of UGS class (in priority queue) are served in the next slot of frame allocation time. Thereafter, the RTPS connections are served by ordering them according to queue size information. In this case, $\sum_{i=1}^n B_{RTPS}$ are the total request with early deadline in RTPS class. The request which are waiting for a long time in a priority queue of RTPS are served

according to their weight factor. The remaining request of RTPS class whose dead line is remaining are served in the next time slot of frame allocation. In our algorithm, we have fixed the percentage of bandwidth allocated to each class of traffic. If there is no higher priority traffic flows, the reserved bandwidth can be utilized by the lower class traffic including the intermediate queue.

4.1 Example

We have considered an WiMAX network connected to subscriber and mobile nodes shown in Fig. 2 to illustrate the working principle of APBS algorithm. There are just one base station and two subscriber stations that are connected to the base station. There are four nodes/end users, namely N0, N1, N2 and N3. The nodes N0 and N1 are connected directly to the base station. The N2 and N3 are connected via the subscriber stations. We have currently four requests for each class with bandwidth demand in kilo bytes. The traffic pattern are shown in Table 1 (and are graphically depicted in Fig. 1).

We use the following example to explain our algorithm. We have set the frame size as 100 KBps (kilo bytes per second) and frame duration as 0.5 s.

This frame allocation in Table 2 shows very simple distribution of the frame size. An important thing to watch here is that the UGS class is allocated just 60% of the total frame size despite being the highest priority class. Hence, other classes also get their requests served.

Fig. 1 Scheduler for uplink at base station

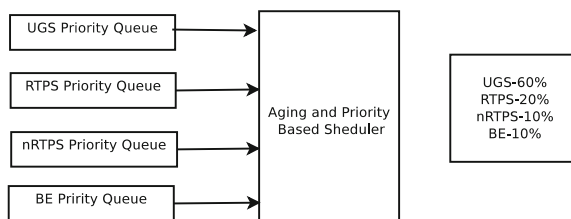


Fig. 2 The WiMAX network structure

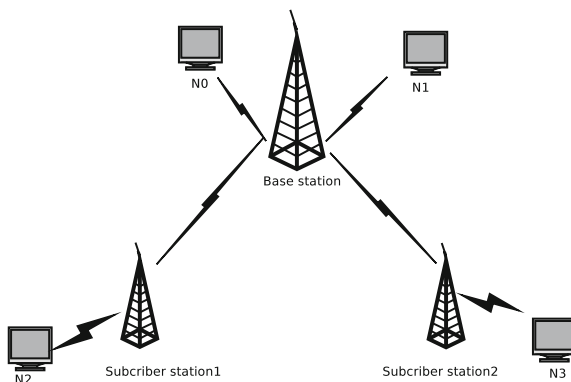


Table 1 Bandwidth demanded from different class of traffic

Request	Node	$Class_{type}$	$B.requested(KB)$	S.time(s)	Duration(s)
R_0	0	UGS	120	0	1
R_1	1	RTPS	50	0.5	2
R_2	2	RTPS	36	0	1
R_3	3	nRTPS	27	1	2
R_0	0	nRTPS	10	0.5	1
R_1	1	BE	3	0	1
R_2	2	BE	8	1	2
R_3	3	BE	5	0.5	1

Table 2 After running the scheduling algorithm, time $t = 0s$

Request	$Class_{type}$	$B.requested(KB)$	Age	Deadline	Weight	B.allocated
R_0	UGS	120	0	1	0.25	60
R_2	RTPS	36	0	1	0.25	20
R_5	BE	3	0	1	0.25	3

Table 3 Frame allocation for time $t = 0.5s$

Request	$Class_{type}$	$B.requested(KB)$	Age	Deadline	Weight	B.allocated
R_0	UGS	60	0.5	0.5	0.5	60
R_2	RTPS	16	0.5	0.5	0.5	16
R_1	RTPS	50	0	2	0.33	4
R_4	nRTPS	10	0	1	0.25	10
R_4	BE	5	0	1	0.2	5

Table 4 Frame allocation for time $t = 1s$

R	$Class_{type}$	$B.requested(KB)$	Age	deadline	Weight	B.allocated
R_1	RTPS	46	0.5	1.5	0.3	46
R_3	nRTPS	27	0	2	0.167	27
R_6	BE	8	0	2	0.1672	8

This frame shows the competitive frame allocation among requests of same class. Here, we can see that there are two requests from RTPS class. However, the one with greater weight value gets served first. The frame allocation at $t = 0.5$ as shown in Table 3. There is also competition for frame allocation between traffic requests from same class. Here, we can see that there are two requests from RTPS class. However, the one with greater weight value gets served first.

This frame allocation in Table 4 shows the reallocation property among the classes in terms of frame size for each class. We can observe that 46 KBps of RTPS request is satisfied despite the limit for RTPS request being 20 KBps initially.

It is because of the reallocation of frame size. Hence, RTPS is able serve up to 50 KBps (initially limited to 20 KBps) request in the above frame, and nRTPS is able to serve up to 27 KBps (initially limited to 10 KBps). All requests are served in an efficient manner.

$$RTPSF.size = \left[\frac{RTPS}{Remaining} \frac{F.size + RTPS}{classes} \frac{frame\%}{F.size\%} * F.size \text{ leftout} \right]$$

$$\text{Example: RTPS frame size} = 20 + \left[\frac{20}{20+10+10} \times 60 \right] = 50 \text{ KB.}$$

$$\text{Similarly for nRTPS} = 10 + \left[\frac{10}{20+10+10} \times 60 \right] = 25 \text{ KB.}$$

Also after RTPS leaves out 4 KB:

nRTPS new size = $25 + \left[\frac{10}{10+10} \times 4 \right] = 27 \text{ KB}$. RTPS size actually reflects the extra bandwidth available for RTPS class, when no higher class traffic request are made.

Algorithm 1: Ageing and Priority-Based Scheduling (APBS) in WiMAX network

```

1 Input:  $P$  is a instance of of traffic requests. The  $P$  consists of
    $[S, B.requested, Class_{type}, start_{timeofrequest}, request.duration]$  and  $BS = [ss_1, \dots, ss_n]$  attached
2 Output: Magazing the network throughput of given network.

3 step 1: Allot all subscriber stations to Nearest base station.
4 step 2: for  $i = 1$  to  $stime$  do
5     step 2.1: for  $framestart$  to  $frameend$  do
6         step 2.2:  $current_{timeofsystem} = i + j$ 
7         step 2.3: if ( $request.startTime \leq current_{timeofsystem}$ )
           and( $start_{timeofrequest} + request.duration \geq current_{timeofsystem}$ ) and
           ( $B.requested! = B.maxAllocated$ ) then
8             step 2.4: Add the request to list  $L_{req}$ 
9             step 2.5: Check to which node the request belong, assign the node object to the
           request.
10            step 2.5.1: Get the current co-ordinates of the nodes.
11            step 2.5.2: Assign nodes to nearest station(whether BS/SS) on the distance and
           range.
12        end
13    step 3: for  $k = 1$  to  $BS_k$  do
14        step 3.1: for each request of  $L_{req}$  do
15            step 3.2: Arrange them in queues in the base station.
16            /* where Priority Queue of all classes. */
17        end
18        3.3: call scheduling(Priority queues,  $current_{timeofsystem}$ );
19        step 3.4: Proceed with the request in the queue. Add the scheduled request to
           get final request of all base station
20    end
21 end
22 end

```

Algorithm 2: Scheduling algorithm in WiMAX network

```

1 Input: A Priority Queues of all classes  $Class_{types}$ 
2 Output: Scheduling the bandwidth among the higher to lower priority classes to avoid
   starvation of lower priority classes.
3 step 1: for  $i = 1$  to  $|Class_{types}|$  do
4     /* For each class in (UGS, RTPS, nRTPS, BE) */
5     step 1.1: for  $req_j$  to  $class_i$  do
6         Calculate weight of request using the formula.
7         step 1.2:  $Weight_j = (\alpha \times (age + 1)) / (deadline + 1)$ .
8         step 1.3:  $age_j = current_{timeofsystem} - req_j.star_{timeofrequest}$ 
9         step 1.4:  $deadline_j = req_j.End_{timeofrequest} - req_j.currentstart_{timeoftherequest}$ 
10        step 1.5: Schedule request according to weight in priority queue.
11    end
12 end
13 step 2: Initialize all size of  $class_i$ .
14 step 2.1: while  $req_j$  in priority queue  $class_i$  do
15     step 2.2: Check allocable request frame size for the class.
16     /* For each class in (UGS, RTPS, nRTPS, BE) */
17     step 2.3: if  $(req.Allocation_i \leq size.class_i)$  then
18         step 2.3.1:  $size.class_i -= B.requested_i$ 
19         step 2.3.2:  $B.allocation_i += B.requested_i$ 
20     else
21         step 2.4:  $B.allocation_i += size.class_i$ .
22         step 2.5:  $size.class_i = 0$ ;
23     end
24 end
25 end
26 step 3: if  $size.class_i > 0$  then
27     step 3.1: Distribute the remaining bandwidth among all classes.
28      $RTPSF.size = \left[ \frac{RTPSF.size + RTPSframe\%}{RemainingclassesF.size\%} * F.size \quad leftout \right]$ 
29 end

```

5 Complexity Analysis

By algorithmic analysis, the total complexity of the algorithm is $O((no.BS) * ((no.R_N) * (ss_n) + (no.R_N) * \log(no.R_N)))$. The algorithm is efficient and is feasible. We also notice that the complexity of the algorithm is eventually independent of the number of subscriber/relay stations. The ABPS algorithm based on multiple queues performs much better than the traditional first come first serve (FCFS) algorithm. We assume that number of requests is going to be much greater than number of base stations and subscriber stations. Hence, complexity reduces to $O(BS * (ss_n * R_N + R_N \log R_N))$, where BS is the number of base stations, ss_n is the number of nodes/end users, and R_N is the number of requests generated.

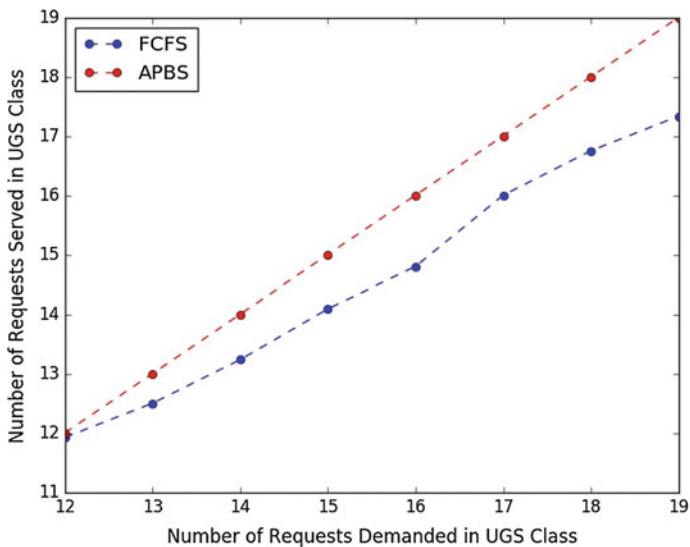


Fig. 3 Relationship of total number of requests demanded versus total number of requests served in UGS class

6 Results

In this section, the simulation results are observed under several scenarios by varying the traffic request of different classes. Total traffic request generated by subscriber stations under categories UGS:RTPS:nRTPS and BE are in ratio of 20:20:60, and requests are generated between the range of 64 KBps–128 KBps, 30–64 KBps and 1 KBps–30 KBps, receptively.

Figures 3, 4, 5 and 6 show the results of APBS with FCFS scheduling algorithm. They demonstrate the relationship between the number of the number of requests served in all four class with the number of requests demanded in all four class of traffic. The proposed algorithm outperforms the existing algorithm FCFS.

Figures 7, 8, 9 and 10 depict the results of APBS with FCFS scheduling algorithm. The proposed algorithm performed better in terms of serving the total bandwidth demanded in all classes compared to the existing algorithm FCFS.

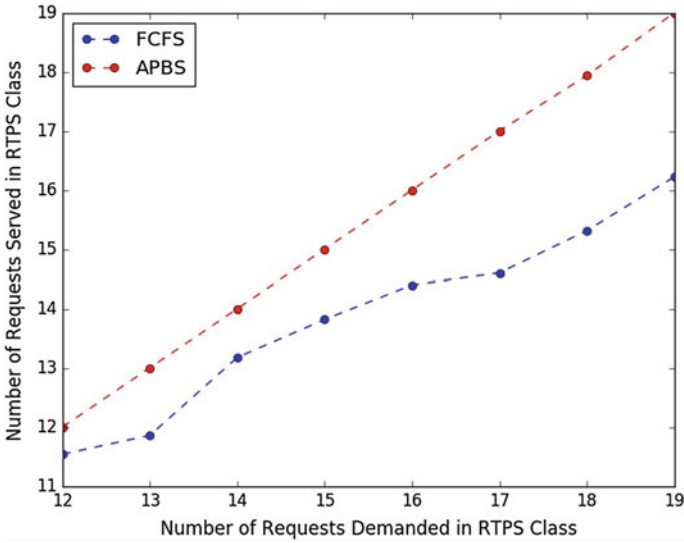


Fig. 4 Relationship of total number of requests demanded versus total number of requests served in RTPS

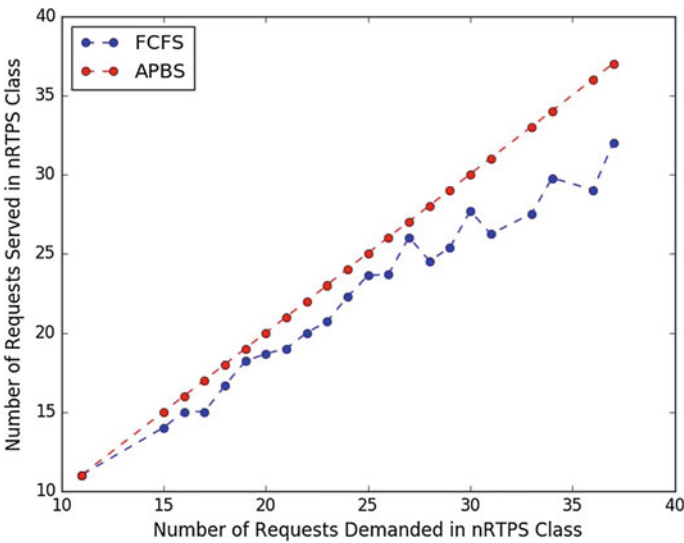


Fig. 5 Relationship of total number of requests demanded versus total number of requests served in nRTPS class

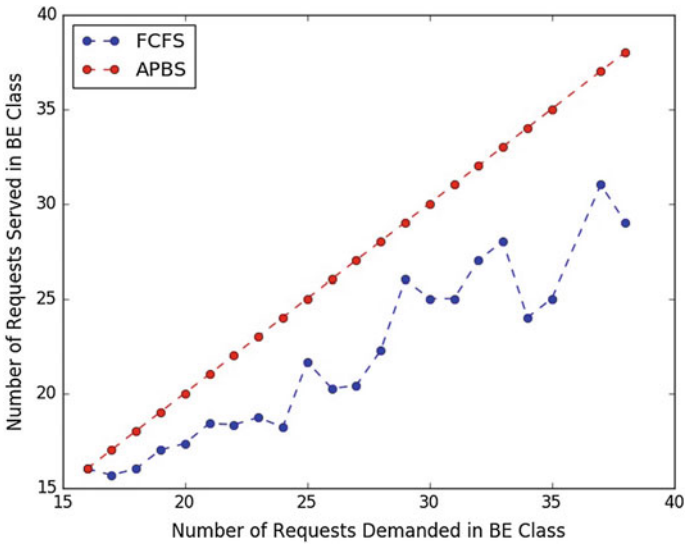


Fig. 6 Relationship of total number of requests demanded versus total number of requests served in BE class

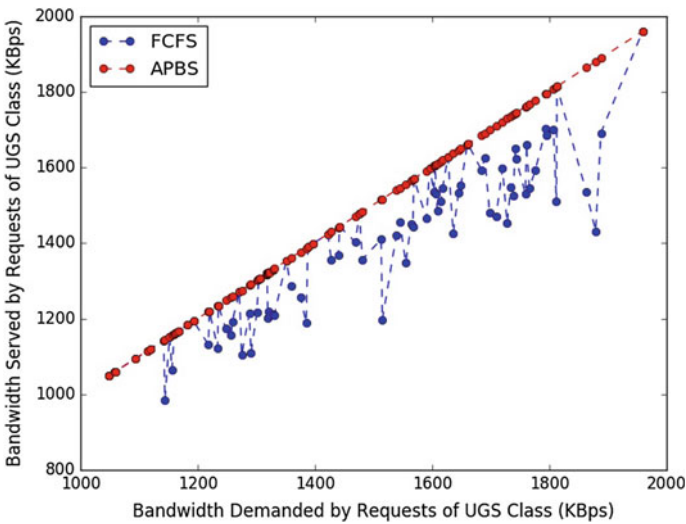


Fig. 7 Relationship of total bandwidth demanded versus total bandwidth served in UGS class

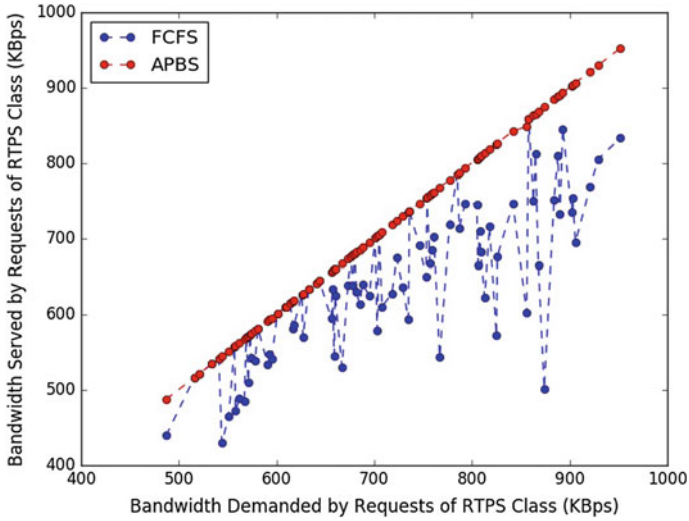


Fig. 8 Relationship of total bandwidth demanded versus total bandwidth served in RTPS class

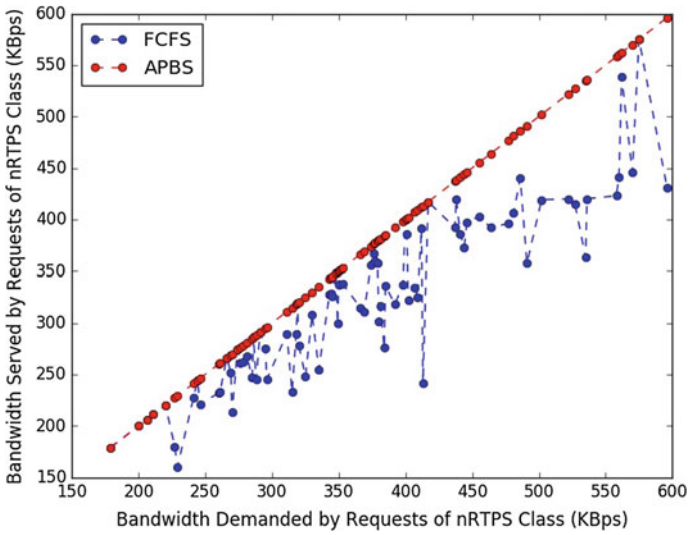


Fig. 9 Relationship of total bandwidth demanded versus total bandwidth served in nRTPS class

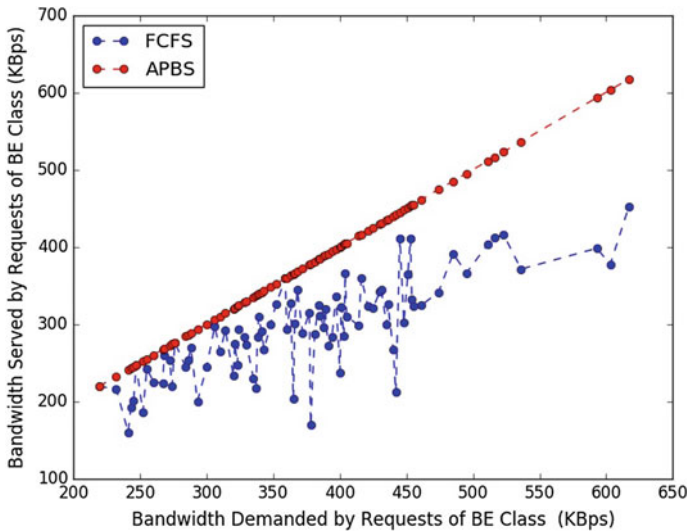


Fig. 10 Relationship of total bandwidth demanded versus total bandwidth served in BE class

7 Conclusion and Future Work

In this work, we have designed algorithm, Ageing and priority-Based Scheduling (APBS) to eliminated the bandwidth starvation due to the higher class monopolizing the lower class traffic. We have also significantly reduced starvation of requests, particularly of the less priority classes (like best effort traffic). The proposed algorithm can be extended to dynamic traffic demands in the hybrid networks in future.

References

1. Chen, J.C.: Efficient batch-arrival queueing model for QoS-aware packet scheduling in WiMAX networks. In: 2010 IET International Conference on Frontier Computing. Theory, Technologies and Applications, pp. 221–226 (2010)
2. Alsahag, A.M., Ali, B.M., Noordin, N.K., Mohamad, H.: Fair uplink bandwidth allocation and latency guarantee for mobile wimax using fuzzy adaptive deficit round robin. *J. Netw. Comput. Appl.* **39**, 17–25 (2014)
3. Cicconetti, C., Erta, A., Lenzi, L., Mingozzi, E.: Performance evaluation of the IEEE 802.16 MAC for QoS support. *IEEE Trans. Mob. Comput.* **6**(1), 26–38 (2007)
4. Yadav, A., Vyavahare, P., Bansod, P.: Review of WiMAX scheduling algorithms and their classification. *J. Inst. Eng. (India): Ser. B* **96**(2), 197–208 (2015)
5. Sagar, V., Das, D.: Modified EDF algorithm and WiMAX architecture to ensure end-to-end delay in multi-hop networks. In: TENCON IEEE Region 10 Conference, pp. 1–6 (2008)
6. Fathi, M., Rashidi-Bajgan, S., Khalilzadeh, A., Taheri, H.: A dynamic joint scheduling and call admission control scheme for IEEE 802.16 networks. *Telecommun. Syst.* **52**(1), 195–202 (2013)