AI Based Qos Scheduler for WIMAX

D. David Neels Pon kumar
Department of ECE
Sardar Raja College of Engineering and Technology, Anna University
Tamilnadu, India

Dr. K. Murugesan
Department of ECE
Bharathiar Institute of Engineering for women, Anna University
Tamilnadu, India

G. Annapoorani
Department of ECE
Sardar Raja College of Engineering and Technology, Anna University
Tamilnadu, India

ABSTRACT

Worldwide Interoperability for Microwave Access (WiMAX) is one of the most familiar broadband wireless access technologies that support multimedia transmission. IEEE802.16 Medium Access Control (MAC) covers a large area for bandwidth allocation and Quality of Service (QoS) mechanisms for various types of applications. Nevertheless, the standard lacks a MAC scheduling algorithm that has a multi-dimensional objective of satisfying QoS requirements of the users, maximizing channel utilization while ensuring fairness among users. So a novel Priority based Scheduling Algorithm using Artificial Intelligence that addresses these aspects are proposed. The initial results show that maximum channel utilization is achieved with a negligible increment in processing time while keeping the priority intact.

General Terms
Algorithms, Performance

Keywords
WiMAX, Fuzzy, Priority, Scheduling Algorithm

1. INTRODUCTION
Recently, the IEEE 802.16 standards (e.g., 802.16- 2004, 802.16e) [1] are noticed to a greater extent and is a viable alternative to the traditional wired broadband techniques due to its cost efficiency. It is envisioned that WiMAX will provide the last mile internet access to every residential user. A high level of QoS and scheduling support is one of the interesting features of the WiMAX standard. These service-provider features are especially valuable because of their ability to maximize air-link utilization and system throughput, as well as ensuring that Service-level agreements (SLAs) are met [6]. QoS is enabled by the bandwidth request and grant mechanism between various subscriber stations and base stations**.** Primarily there are five buckets for the QoS (UGS, rtPS, ertPS, nrtPS, and BE) to provide the service-class classification for video, audio, and data services. The service scheduler provides scheduling for different classes of services for a single user. This would mean meeting SLA requirements at the user level.
The five service flows are explained below:

1) Unsolicited grant service (UGS): This service can provide guaranteed data throughput and latency.
2) Real-time polling service (rtPS): The minimum reserved rate and the latency are guaranteed in this application.
3) Enhanced Real-time polling service (ertPS): It especially concentrates on real time Voice over IP.
4) Non-real-time polling service (nrtPS): This service tolerates delay while streaming variable-sized data packets.
5) Best effort (BE): The channel access mechanism of this service is based on the contention and provides no QoS guarantees.

Even though there are lots of conventional scheduling algorithms they are not meeting all the required QoS parameters. The performance affecting parameters like fairness, bandwidth allocation, throughput and latency are studied and found out that none of the algorithms perform effectively for both fairness and maximum bandwidth utilization simultaneously [2]. So a decision has been made to optimize those two parameters by using an algorithm based on Artificial Intelligence (AI). This paper is organized as follows: Section 2 describes the related work. Section 3 and 4 explain the proposed scheme. Section 5 and 6 shows the performance of WiMAX using the newly proposed scheduling algorithm and the conclusions in section 7.

2. PREVIOUS WORK
In [6], the authors propose a hybrid of Earliest Due Date (EDD) and Weighted Fair Queue (WFQ). In EDD, all the arriving packets get a deadline stamp and are scheduled according to the increasing order of deadlines. The algorithm intends to serve the real time traffic first and only if real time buffer is empty will they consider BE traffic. This will certainly lead to starvation. In [7], the authors consider two types of queues.
The first type is used to schedule data grants for UGS and allocate request opportunities for rtPS and nrtPS. These grants are scheduled in a first in first out (FIFO) manner. Once the first queue type has been served, the scheduler will consider the second type leading to scarcity. The authors in [8] propose an architecture consisting of three schedulers. The first scheduler is concerned with UGS, rtPS and ertPS flows. The second scheduler is concerned with flows requiring a minimum bandwidth mainly nrtPS. The third scheduler is used for BE traffic the third scheduler comes into picture only when the first two schedulers have become free. In [9], the suggested uplink scheduling algorithm is Weighted Round Robin (WRR) with GPSS grant mode. The duration of contention slots and uplink data slots are dynamically distributed according to bandwidth requirements. The authors did not comment on what weights to use for WRR

scheduling or BS downlink scheduling. In [10], the authors suggest downlink. bandwidth allocation algorithms based on flow type and strict priority from highest to lowest - UGS, rtPS, ertPS, nrtPS and BE. Here an Opportunistic fair scheduling was used. Here BE traffic is served whenever an opportunity is available, but for most of the time BE starves for bandwidth.

3. FUZZY SCHEDULER

3.1 The Fuzzy System

Basically the fuzzy system consists of four blocks, namely, fuzzifier, defuzzifier, inference engine, and fuzzy knowledge base. The following section explains the working of a general fuzzy system.

3.1.1 Fuzzification

The first step is to take the inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. A fuzzy set A in the universe of discourse U is a set of ordered pairs $\{(x_1, \mu A(x_1)), (x_2, \mu A(x_2)), \ldots, (x_n, \mu A(x_n))\}$, where $\mu A : U \rightarrow [0, 1]$ is the membership function of the fuzzy set A and $\mu A(x_i)$ indicates the membership degree of x_i in the fuzzy set A.

3.1.2 Fuzzy inference process

If a fuzzy system has n inputs and a single output, its fuzzy rules Rj can be of the following general format. (Rj) If X_1 is A_1 j , X_2 is A_2 j , X_3 is A_3 j , . . ., and X_m is A_mj, then Y is Bj . The variables $X_i\{i = 1, 2, 3, \ldots, n\}$ appearing in the antecedent part of the fuzzy rules Rj are called the input linguistic variables, the variable Y in the consequent part of the fuzzy rules R_j is called the output linguistic variable. The fuzzy sets A_ij are called the input fuzzy sets of the input linguistic variable Xi and the fuzzy sets Bj are called the output fuzzy sets of the output linguistic variable Y of the fuzzy rules Rj .

3.1.3 Aggregation of all outputs

Since decisions are based on the testing of all of the rules, the rules must be combined in some manner in order to make a decision. Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set.

3.1.4 Defuzzification

As much as fuzziness helps the rule evaluation during the intermediate steps, the final desired output for each variable is generally a single number. However, the aggregate of a fuzzy set encompasses a range of output values, and so must be defuzzified in order to resolve a single output value from the set. The most popular defuzzification method is the Centroid calculation, which returns the centre of area under the curve. By Centroid method of defuzzification, the crisp output η is calculated using the formula,

$$\eta = \frac{1}{\sum \mu_{x1\cdots xn}^{\text{output}}(y)} \sum y \mu_{x1\cdots xn}^{\text{output}}(y)$$

…… (Eqn 3.1)

where y is the centre point of each of the output membership function in the output fuzzy set Bj and μ output $x_1 \cdots x_n(y)$ is the strength of the output membership function .

3.2 The Primary fuzzy scheduler

The incoming requests in the WiMAX have different variables that play a key role in setting the priority of that particular request. The variables are Expiry Time, Waiting Time, Queue Length, Packet Size and Type of Service. In the proposed fuzzy scheduler two different stages namely the Primary Scheduler, FS1 and the Dynamic Scheduler, FS2 are used. This proposed scheduler is named as Dynamic Fuzzy based Priority Scheduler (DFPS) which uses four inputs namely, Expiry time (E), Waiting time (W), Queue length (Q), Packet size (P) and one output, Priority index as shown in Figure 1. Here, the process is considered as multiple input and single output (MISO) system. The fuzzy rule table is created based on the membership functions (Figure 2) that are carefully designed as explained in table 1. The linguistic terms associated with the input variables are low (L), medium (M) and high (H). Triangular membership functions are used for representing these variables except for the high data rate where a trapezoidal function is used. The bases of functions are chosen so that they result in optimal value of performance measures For the output variable, priority index, five linguistic variables are used. Only triangular functions are used for the output. This illustration was designed using the fuzzy tool available in the MATLAB. For illustration the ninth rule is interpreted as "If packet size is high and queue length is high, then priority index is high".

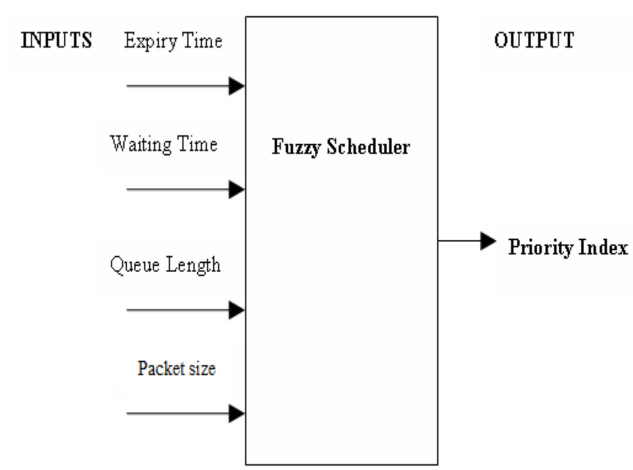
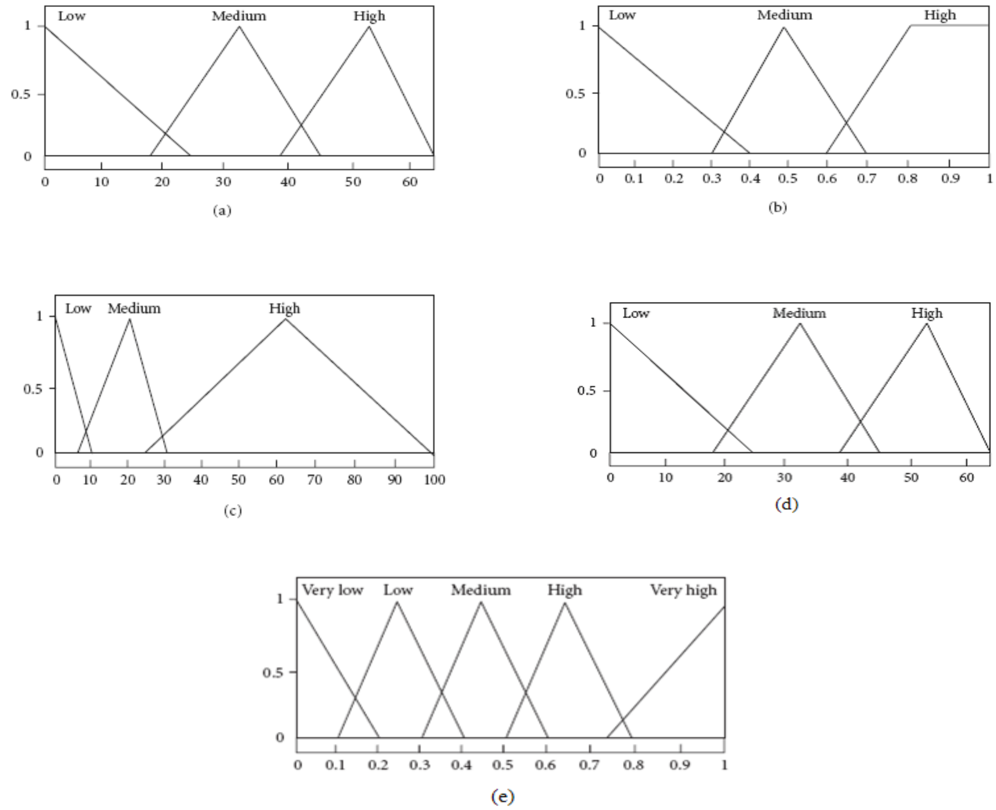


Figure 1. Proposed Primary Fuzzy Scheduler



(a) Expiry time (in sec) (b) Packet size (in Kbytes) (c) Queue length (in bytes)

(d) Waiting time (in sec) (e) Priority Index

Similarly, the other rules are framed. The priority index, if high, indicates that the packets are associated with the highest priority and will be scheduled immediately. If the index is low, then packets are with the lowest priority and will be scheduled only after higher priority packets are scheduled.

3.3 Dynamic Fuzzy Scheduler

For a dynamic scheduler, the output of the primary scheduler is given as the input. Apart from this input, the type of service variable is also added as shown in Figure 3. A membership function and a Rule base table are created based on the priority index of FS1 and the type of service. The Dynamic Fuzzy Rule Base is shown in table 2. This table is carefully designed by taking into consideration of the type of service. As there are five different types of classes the priority levels are set to five different levels starting from Very High (VH), High (H), Medium (M), Low (L) and Very Low (VL). To illustrate any rule, consider the first column contents. The Priority Index of the Primary Scheduler may be from VH to VL.

If the type of service is UGS then that request must be given higher level priority than the other type of services even if the Primary Scheduler FS1 allots them higher priority indices. The final priority index is referred as η which is the standard notation used in the literature.

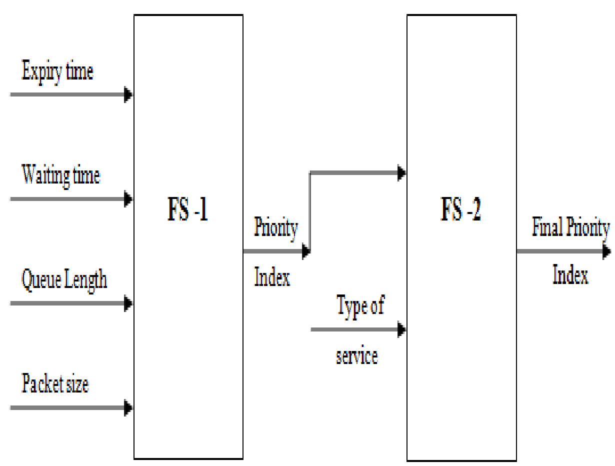


Figure 3. Dynamic Fuzzy scheduler

Table 1. Fuzzy Rule Base

(a) Expiry Time Vs Waiting Time

Expiry Time	**Waiting Time**		
	L	**M**	**H**
L	H	L	L
M	M	H	L
H	L	M	H

(b) Packet size Vs Queue length

Packet Size	**Queue Length**		
	L	**M**	**H**
L	H	M	M
M	L	H	M
H	L	L	H

(c) (a) Vs (b)

(a)	**(b)**		
	L	**M**	**H**
L	VL	L	M
M	L	M	H
H	M	H	VH

Table 2. Dynamic Fuzzy Rule Base

Priority	**UGS**	**rtPS**	**ertPS**	**nrtPS**	**BE**
VL	VH	L	L	VL	VL
L	VH	M	L	L	VL
M	VH	H	M	L	L
H	VH	H	M	M	L
VH	VH	VH	H	M	L

Algorithm 1:- Setting up the priority
Input: Expiry Time, Waiting Time, Packet Size, Queue Length, Type of Service
Assumptions: The above said first four inputs are in 3 scales (High, Medium, Low) and last input is in 5 scales (Very High, High, Medium, Low, Very Low).
Output: Highest Priority Request

For i=1 to n do

a. Compare Expiry Time and Waiting Time which arrives at Priority index 'a'.
b. Compare Packet Size and Queue Length which arrives at priority index 'b'.
c. Compare 'a' and 'b' (3 scale output) which arrives at Intermediate Priority (5 scale output).
d. Compare Intermediate Priority with Type of Service which arrives at Final Priority Index.
e. Arrange the request in descending order.

4. SCHEDULING USING ANN

The next step is scheduling of the prioritized input received from the DPFS. A neural network is a massively parallel-distributed processor that has a natural propensity for storing experiential knowledge and making it available for use.
An ANN have the following features:

- Adaptive Learning
- Self-Organization
- Real Time Operation
- Fault Tolerance via Redundant information coding

Despite the diversity of network paradigms, nearly all are based upon this configuration. A set of inputs labeled $x_{1,} x_{2......} x_n$ is applied to the artificial neuron. These inputs collectively referred to as the vector X correspond to the signals into the synapses of a biological neuron. Each signal is multiplied by an associated weight $w_{1,} w_{2...}$ wn before it is applied to the summation block, labelledΣ. Each weight corresponds to the "strength" of a single biological synaptic connection. The set of weights is referred to collectively as the vector W. The summation block, adds all of the weighed inputs algebraically, producing an output that we call NET. This may be compactly stated in vector notation as follows:

NET = XW

NET = $x_1 * w_1 + x_2 *w_2 + x_3*w_3+ \dots + x_n *w_n$

4.1 Proposed ANN

The proposed ANN is shown in Figure4. It consists of three layers. The first layer is the input layer and the second layer is the modified form of Kohonen layer. The final layer is the modified form of Grossberg layer. The proposed ANN deals with the efficient allocation of the available bandwidth based on the Priority Index set by the DFPS with a measure of fairness to all the service class es. The input layer receives the prioritized outputs from the DFPS. These inputs are organized in the order of their priority. Now the output of this layer is given as the input to the modified Kohonen Layer.

The modified Kohonen layer is used to predict whether the given input is within the threshold value. Depending on the availability of the channel bandwidth the threshold value is set. If the incoming request is below the threshold value then that request is forwarded to the next layer, the Grossberg layer. If not, that request is rejected, it happens on extreme circumstances. In the Grossberg layer, the inputs are summed up and it calculates how many requests can be granted within the threshold value. DFPS output is given to the input layer and according to the weight it is processed and given to layer 1 which is Kohenon layer, where it checks the request with the threshold value if it is accessible request is granted. The next layer is Gross berg layer where the inputs are calculated and within the threshold requests are granted. The action of each neuron in the Gross berg layer is to output the value of the weight that connects it to the single nonzero Kohonen neuron.

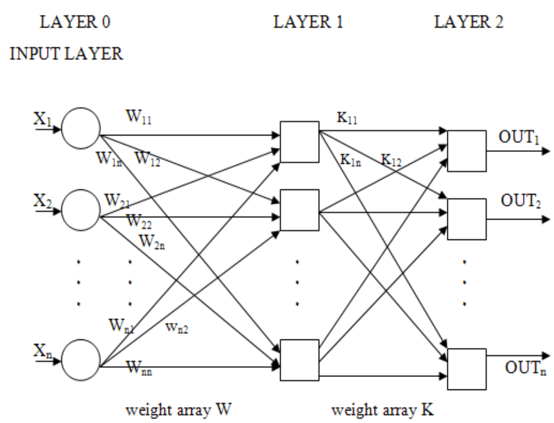


Figure 4. Proposed ANN

Algorithm 2:- Scheduling using ANN

Input: Prioritized Request, Threshold Value

Output: Scheduling the request

For i=1 to n do

In kohonen layer

a. **If** input < threshold, send to Grossberg layer **else** the request is rejected.

In Grossberg layer

b. Compare Sum of bandwidth of requests with threshold

   If possible, set Sum as bandwidth of the request

   Else go for the next request.

c. Sum = Sum + Bandwidth

d. **If** threshold > Sum, Set the tag of request to not possible and store the request number as limit

   Else select low priority request starting from bottom

e. Repeat steps b and c

f. **If** threshold > Sum, tag the lower priority request as possible and select the next low priority request

   Else Tag the low priority request as not possible and select the next low priority request. Then, go to step g.

g. **If** Low priority request number = Limit, stop
   Else go to step e.

5. PERFORMANCE METRICS

5.1 Effective channel utilization

The algorithm must utilize the channel efficiently. This implies that the scheduler should not assign a transmission slot to a session with a currently bad link since the transmission will simply be wasted.

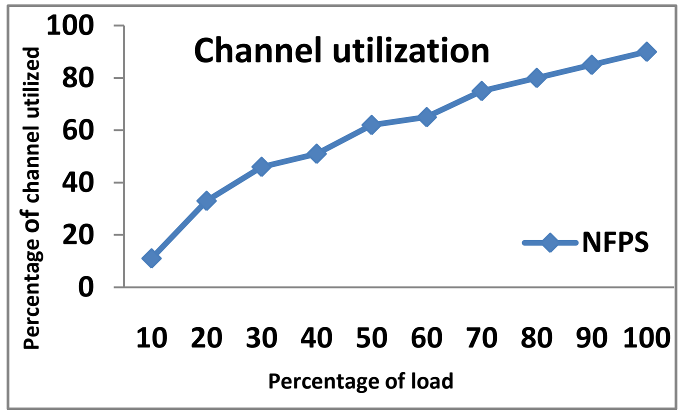


Figure 5. Graph showing percentage of channel utilized using NFPS Algorithm

5.2 Fairness

The scheduling algorithm must provide fairness to all the requests with different quality of service classes. The channel starving lower priority BE requests and nrtPS requests must be satisfied too leading to fairness.

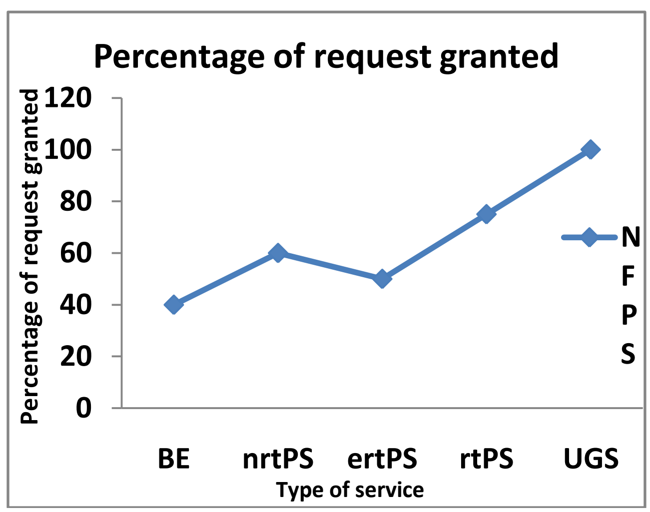


Figure 6. Graph showing percentage of request granted for different types of services using NFPS Algorithm

5.3 Processing time

The algorithm must be able to provide delay bound guarantees for individual sessions in order to support delay-sensitive applications that largely depend on the processing time.

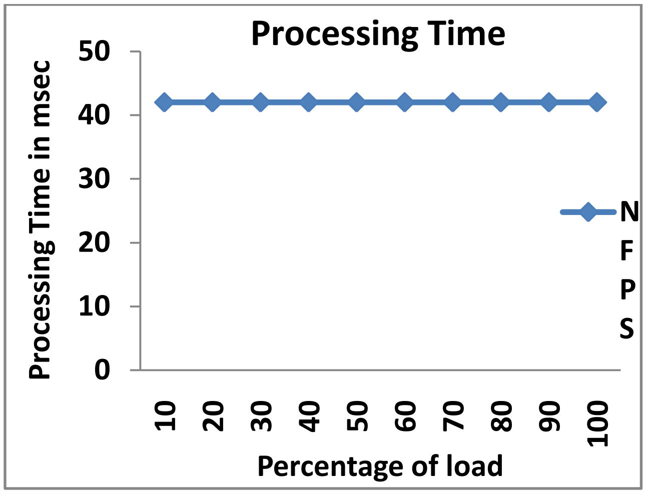


Figure 7. Graph showing processing time using NFPS Algorithm

6. PERFORMANCE COMPARISON

The Performance of the proposed NFPS Algorithm is studied under various metrics. Firstly the Channel utilization aspect is analyzed for proposed NFPS Algorithm versus the conventional scheduling algorithms. Then the percentage of requests granted versus the type of service which reveals the amount of fairness obtained while keeping the priority intact is studied and compared with the conventional scheduling algorithms. Here the study was carried out for different set of requests. Finally the processing time was calculated and compared with the conventional scheduling algorithms.

6.1 Channel Utilization

The channel utilization is calculated. The Figure 5 clearly shows the amount of channel utilized by our proposed NFPS Algorithm. It begins from 10% for 10% of load to almost 90% for full load. So as the number of requests increases the channel utilization also increases. It is inferred that as the requested bandwidth nears the total load, the percentage of channel utilization increases. It is understood from the Figure 8 that the DFQ utilizes almost 85%, WFQ utilizes as high as 75% and OFS utilizes 80% for the same set of requests. So the comparisons clearly show that there is under utilization of resources in the existing algorithms. It is also inferred from the graph that at no point the conventional algorithms out performs our proposed NFPS algorithm. Hence it is imperative that maximum channel utilization is achieved in our proposed NFPS algorithm. Generally it lies in the zone of 90% to 95%.So there is no point in pondering of under utilization in our algorithm.

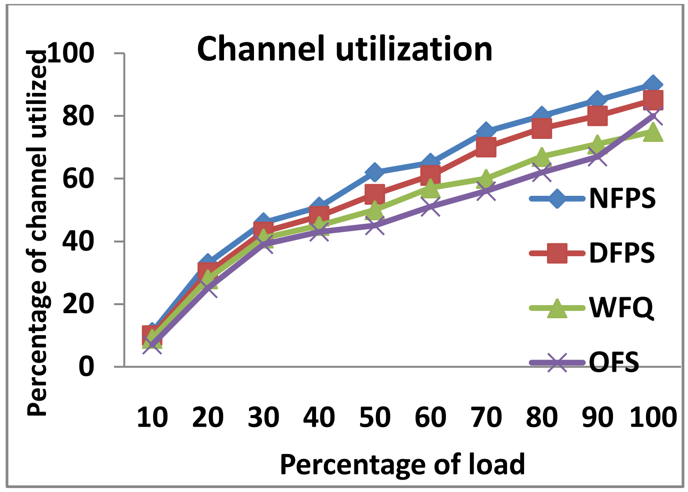


Figure 8. Graph showing the comparison of percentage of Channel utilized using NFPS Algorithm versus conventional Algorithms.

6.2 Fairness analysis

In the Figure 6 all the requests of UGS i.e. 100% are granted. 75% of the requests of the rtPS are granted. But in the case of ertPS 50% of the requests are granted. Even though nrtPS and BE have lower priority 60% and 40 % of their requests are granted respectively. It shows that the UGS traffic of WiMAX is handled first and is scheduled without any trouble. This satisfies the basic rule of IEEE 802.16 standard. Then a portion of rtPS and ertPS are also granted depending on the availability and the fuzzy rule base. But the success of this Algorithm is the granting of requests from the lower priority service classes (nrtPS and BE) consistently. Hence here the priority is kept intact while the once channel starving lower priority service classes are been taken care of leading to fairness. The Figure 9 shows the comparison of the percentage of requests granted for the various types of service classes for different conventional Scheduling Algorithms with the proposed NFPS Algorithm. The graph reveals that the Shortest Job First (SJF) algorithm does not consider priority at all and on sight it violates WiMAX basic rule and also there is no provision for fairness. It is imperative that the First Come First Serve (FCFS) does not care about priority or fairness but it grants the request on first come first serve basis even though it is not shown in the graph. It is inferred from the graph that Weighted Fair Queuing (WFQ), and Opportunistic Fair Scheduling (OFS) [6] - [10] that aims at fairness as indicative of the name grants all the requests of UGS service class. But they grant only 5% and 10% of the least priority one the BE service class respectively where as our proposed Algorithm grants as high as 40% of the requests.

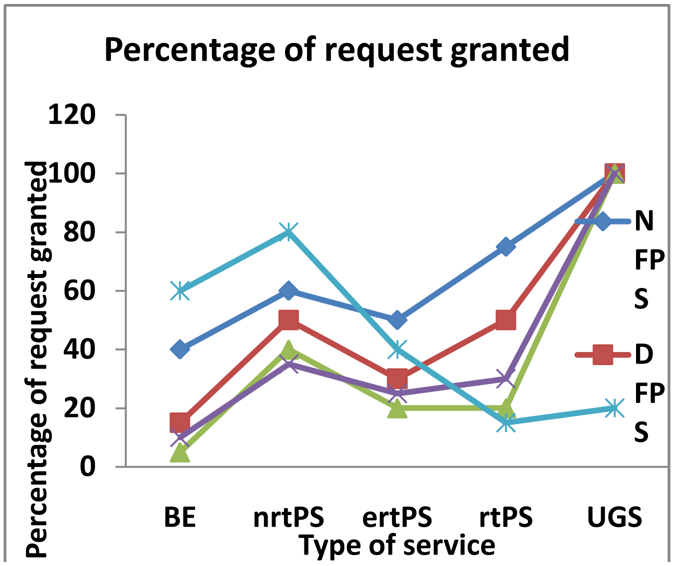


Figure 9. Graph showing the comparison of percentage of request granted for different types of services using NFPS versus conventional Algorithms

Even though there is a little amount of fairness in WFQ and OFS algorithms most of the time the BE service class requests must starve for resources. Hence it is inferred that our NFPS algorithm improves fairness to an extent while keeping the priority intact.

6.3 Processing Time

Eventhough the proposed algorithm is way ahead in fairness, priority and channel utilization, the next aspect the processing time was studied. Figure 7 shows that the procesing time for the proposed algorithm to grant a full load traffic and for lighter loads it was 42 milliseconds. But for multimedia applications using Internet permits delays upto 400 milliseconds as acceptable one. So as for as quality is concerned it is not on the wrong side but very much on the highly acceptable grounds. Figure 10 visualises the processing time under full load traffic for conventional algorithms. It is seen that WFQ needs 27 milliseconds to grant full load of requests and OFS needs milliseconds and SJF 17 miliseconds. It infers that the conventional algorithms processs the requests much faster than the proposed algorithm. Therefore it is understandable that this novel scheduling algorithm is bit slower than the traditional scheduling algorithms but the fairness and channel utilization it provides overwhelms that setback as this processing time is well within the acceptable standards of streaming of multimedia over the Internet and Wireless Broadband Networks.

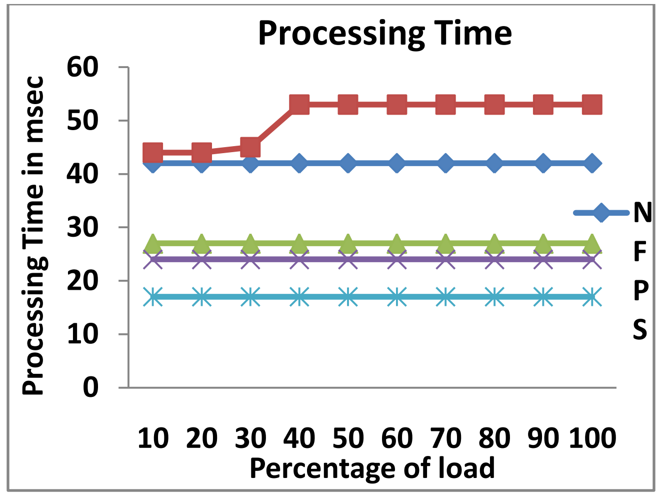


Figure 10. Graph showing the comparison of Processing time using NFPS Algorithm versus the Conventional Algorithms

7. CONCLUSIONS

An Artificial Intelligence based QoS Scheduling Algorithm was designed. The fuzzy section dealt with the priority setting mechanism under uncertainty conditions by taking into consideration of variables such as expiry time, waiting time, queue length, packet size and the type of service for WiMAX requests. Artificial Intelligence section dealt with bandwidth allocation mechanism by considering fuzzy prioritized output as its input. The Simulation results show that a fair amount of fairness is attained while keeping the priority intact. The results also show that maximum channel utilization is achieved with a negligible increment in processing time.

8. REFERENCES

[1] Chao-lieh chen, Jeng-wei lee, Chi-yuan wu and yau-hwang kuo, “Fairness and QoS Guarantees of WiMAX OFDMA Scheduling with Fuzzy controls”, EURASIP Journal on Wireless Communications and Networking, 2009.

[2] Chen,C.-L., “IEEE 802.11e EDCA QoS provisioning with dynamic Fuzzy control and cross-layer interface,” in Proceedings of the 16th International conference on computer communications and networks (ICCCN’07) pp.766-771 Honolulu, Hawaii, USA, August 2007.

[3] David Neels Pon Kumar,D., Murugesan,K., Raghavan,S., “A Dynamic Fuzzy Based Priority Scheduler for Mobile WiMAX”, Proceedings of ICVLSICOM10 pp 189-194, ISBN 978- 93-80043070-8, India, Jan 8-10, 2010.

[4] David Neels Pon Kumar,D., Murugesan,K., Raghavan,S., “Quality of Service Scheduling for IEEE 802.16 Broadband Wireless Access System–An Overview”, International journal “Technology World Malaysia” pp 398 - 404 , ISSN 2180-1614, Oct- Nov 2009.

[5] Guojun Dong, Jufeng Dai School of Electronic Information Engineering, Tianjin University, “An Improved Handover Algorithm for Scheduling Services in IEEE802.16e”, China, IEEE, 2007.

[6] Howon Lee, Taesoo Kwon and Dong-Ho Cho, "An efficient uplink scheduling algorithm for VoIP services in IEEE 802.16 BWA systems", 60th IEEE Vehicular Technology Conference, 2004, vol.5, pp. 3070 – 3074.

[7] Kun yang, Jie zhang, Hsiao-hwa chen, "A Flexible QoS aware service Gateway for heterogeneous wireless networks", *IEEE NETWORK* 2007

[8] Sun,J., Yao,Y. and Zhu,H., "Quality of Service Scheduling for 802.16 Broadband Wireless Access Systems'', IEEE Networks, 2006.

[9] Wong,W.K., Tang,H., Guo,S. and Leung, V. C. M., "A fair scheduling for IEEE 802.16 Broadband Wireless Access Systems," ICC2005, May 16-20, Seoul, Korea.

[10] M. Mehrjoo, M. Dianati, X. Shen, K. Naik "Opportunistic Fair Scheduling for the Downlink of IEEE 802.16Wireless Metropolitan Area Networks", 3rd International Conference on QoS in heterogenous Wired /Wireless Networks, vol.191 2006.