World Applied Sciences Journal 15 (4): 576-583, 2011
ISSN 1818-4952

Bandwidth Allocation in WiMAX Networks Using Learning Automaton

[1]Saeid M. Jafari, [2]Majid Taghipour and [3]M.R. Meybodi

[1]Department of Computer and IT Engineering Qazvin, Iran
[2]University of Applied Science and Technology Urmia, Iran
[3]Department of Electrical and Computer Engineering Amirkabir Tehran, Iran

Abstract: Recent developments on the wireless communication technology have brought much innovativeness to make wireless access networks, e.g. WiMAX systems, to be able to compete with the wired access networks with much more bandwidth. QoS service provisioning is an important issue for deploying such networks. The IEEE 802.16d standard has specified the services should be provided at the medium access control (MAC) layer in WiMAX networks. However, it has left a wide space for research to develop and implement those specified services. In this paper, the issue of differentiated service provisioning will be addressed with the non-real-time polling service in WiMAX systems. The proposed solution has been designed to have an ability to accommodate integrated traffic in the networks with effective scheduling schemes. A series of simulation experiments have been carried out to evaluate the performance of the proposed scheduling algorithm. In our algorithm we introduce a two-phase Learning Automaton queuing (2PLAQ) algorithm tailored for uplink scheduling in the WiMAX network. It aims to strike the balance between delay requirement and fair bandwidth allocation. The results reveal that the proposed solution performs effectively to the integrated traffic composed of messages with or without time constraints and achieves proportional fairness among different types of traffic.

Key words: WiMAX % Scheduling % Learning Automaton % Bandwidth Allocation

INTRODUCTION

WiMAX technology based on the IEEE 802.16 standard [1,2] has a very rich set of features. Indeed, it is a very promising Broadband Wireless Access (BWA) technology. The major attractions of WiMAX systems come from their ability to provide broadband wireless access and potential ability to compete with existing wired systems such as fiber optic links, coaxial systems using cable modems and digital subscriber line (DSL) links with much scalability. The major attractions of WiMAX systems come from their ability to provide broadband wireless access and potential ability to compete with existing wired systems such as fiber optic links, coaxial systems using cable modems and digital subscriber line (DSL) links with much scalability. The WiMAX networks have the capacity to provide flexibility and efficiency to allow coexistence of different types of traffic, such as real-time and multimedia traffic. One important issue in the WiMAX networks design is to support QoS services to different types of traffic. IEEE802.16d standard [1, 2], ratified in June 2004, has specified all the techniques of the WiMAX systems to deliver broadband service in the fixed point-to-point (PTP) or point-to-multipoint (PMP) topologies. And it has proposed a framework for the QoS services for four types of traffic. Unsolicited Grant Service (UGS), real time Polling Service (rtPS), non real-time Polling Service (nrtPS) and Best Effort (BE) QoS classes. UGS supports real-time service flows that have fixed-size data packets on a periodic basis. rtPS supports real-time service flows that generate variable data packets size on a periodic basis. The BS provides unicast grants in an unsolicited manner like UGS. Whereas the UGS allocations are fixed in size. nrtPS is designed to support non real-time service flows that require variable size bursts on a regular basis. BE is used for best effort traffic where no throughput or delay guarantees are provided. Those service classes are defined in order to satisfy different types of Quality of Service (QoS) requirements. However, the IEEE 802.16 standard does not specify the scheduling algorithm to be used. Vendors and operators have to choose the scheduling algorithm(s) to be used. Three types of schedulers must be defined; an uplink and a downlink scheduler both in the Base Station (BS) and just an uplink scheduler for the Subscriber Station (SS) between the different simultaneous connections of the SS.

Corresponding Author: Saeid M. Jafari, Department of Computer and IT Engineering Qazvin, Iran.

In this paper we present a system for packet scheduling that is based on Reinforcement Learning [16]. In our approach, Reinforcement Learning (RL) is used to learn a scheduling policy in response to feedback from the network about the delay experienced by each traffic class and fairness. In our algorithm we introduce a two-phase Learning Automaton Queuing (2PLAQ) algorithm tailored for uplink scheduling in the WiMAX network. It aims to strike the balance between delay requirement and fair bandwidth allocation.

In simulations, we consider both popular Poisson traffic and practical burst traffic that is modeled by the Markov Modulated Poisson Process (MMPP). The simulation results verify the correctness of our analytical models and compare 2PLAQ with other scheduling schemes. It achieves a low drop rate and high throughput while maintaining fairness among different connections at the same time. The paper is organized as follows. In Section II we briefly introduce the Related Scheduling Algorithms. In Section III we present the proposed 2PLAQ algorithm. In section IV we analyze and simulate the Two-phase LAQ Scheduling Algorithm. Finally we invest conclusions in Section V.

Related Scheduling Algorithms

In this Section, We Present Some Schedulers: Several WiMAX scheduling solutions have been proposed. WFQ is proposed and analyzed by an M/G/FQ queuing model in [3]. Although WFQ can guarantee the minimum data rate of the connections, it does not take into consideration the delay constraint. Besides, the time complexity of WFQ scheduling algorithm is high and thus becomes a potential problem for its implementation in the WiMAX network. A two-tier hierarchical architecture is proposed in [4] for WiMAX uplink scheduling. In the higher hierarchy, strict prioritization is used to direct the traffic into the four queues, according to its type. Then, each queue is scheduled according to a particular algorithm, i.e. fixed allocation for UGS, EDF for rtps,

WFQ for nrtps and equal division of remaining bandwidth for BE. Although EDF takes care of the delay requirement of the rtps, grouping multiple rtps connections into one queue fails to guarantee the minimum bandwidth requirement of each individual rtps connection. For example, one rtps connection with tight delay budget may dominate the bandwidth allocation, resulting in starvation of other rtps connections. A similar approach is proposed in [7] and it replaces the strict priority algorithm in the higher hierarchy with Deficit Fair Priority Queue (DFPQ). The basic idea of DFPQ is to use DRR to guarantee the bandwidth allocation of each of the four queues, thus preventing queues with higher priorities from depleting the bandwidth and causing starvation of queues with lower priorities. Analysis model is an important part of the research in scheduling algorithms. In [3], mean delay bound is derived by using the M/G/FQ queuing model, but it is based on WFQ algorithm and does not consider QoS parameters other than MRR. QoS parameters are not considered in [6], where an MMPP model is employed to derive the packet drop rate and characterize uplink rtps and nrtps traffic that share a single First-Come-First-Serve (FCFS) queue. For other scheduling algorithms [4,5,7-12], their performance and effectiveness are demonstrated and compared via simulations only, without in-depth theoretical analysis.

Two-Phase Learning Automaton Queuing (2PLAQ) Scheduling Algorithm: In this section, we first present a two-phase Learning Automaton Queuing (2PLAQ) algorithm that addresses the delay and bandwidth requirements while balancing the fairness and efficiency among different connections. Then, an elegant queuing model is established to derive in theory its performance in terms of packet drop rate and throughput.

Overview of the Two-phase Laq Scheduling Algorithm: The scheduling algorithm aims to meet the QoS requirements of all types of connections. For UGS, the scheduling is straightforward, because a fixed amount of bandwidth is always allocated to each UGS connection during each grant interval, which is determined according to the requested data rate. BE connection has no specific QoS requirement, thus it is not the interest of the study. BE's scheduling can be done via a simple scheme like equal division of the remaining bandwidth among all BE connections. In the following discussion, we focus on rtps and nrtps traffic that has specific QoS requirements on delay and bandwidth. To strike the balance between delay and bandwidth requirements, the proposed scheduling algorithm decouples them and addresses them separately in two phases. In our discussion so far of LAQ, the bandwidth allocation in Phase 1 is based on MRR.

More specifically, each connection is allocated a bandwidth that equals its MRR, unless its total request is lower than its MRR. The remaining bandwidth is then used in Phase 2. Apparently, this is not the only option. In theory, we can make any bandwidth allocation to the two phases. To study the impact of bandwidth allocation between two phases, we introduce a parameter), with 0#)#1 and allocate)*i-MRRi* to Connection i in Phase 1.

There is) for any connection to can provide its requirements. For each connection) should be determined such that to be able to meet these needs. Therefore, we'll have a vector of). The) have to be trained using Learning_Automaton for each connection so that determining) as a selective action, indicates delay and bandwidth of the system. The more we concern about selective action, the less packet loss we'll have and the more fairness we'll get. So, we are going to have both delay and fairness issues handled. If) is small, then) of the reserved bandwidth of each flow is allocated at phase 1 and rest of the bandwidth will remain for next phase. Therefore, small) allocates less bandwidth to first phase results in an increase in packet loss and also falters fairness, on the other hand large amount of), allocates more bandwidth to first phase which leads to reduction in packet loss as well as fairness. At the beginning probability vector contains initial values which are equals. So, chance of selecting any action is equal. After executing *ai*, it will immediately receive reward and based on that probability vector will be updated.

$$p_{t+1,i} = p_t + a*(M_i - R_i)$$

In general, there is a trade-off between the drop rate and fairness when)varies from 0 to 1. For example, a small) indicates more bandwidth allocated to Phase 2, which results in a smaller drop rate but higher unfairness.

Analysis of the Two-Phase LAQ Scheduling Algorithm: In this subsection, we analyze the performance of the proposed LAQ algorithm, in order to gain insight into it and to demonstrate in theory its performance. Similar to our earlier discussions, we focus on the QoS performance of rtps and nrtps connections only in our analysis. Note that, we do not explicitly distinguish rtps and nrtps connections, because they only differ in the amount of delay budget. As to be discussed next, our analytic model is generally applicable to any rtps or nrtps connections with given arrival rates, MRR and delay bound.

Different Queuing Principles: LAQ involves two phases with different queuing principles, i.e. FCFS in Phase 1 and EDF in Phase 2.

Delay Constraint: Each data packet in the queue is associated with a specific delay bound. A data packet is dropped if its waiting time is longer than its delay budget. The delay constraint dramatically increases the analytic complexity.

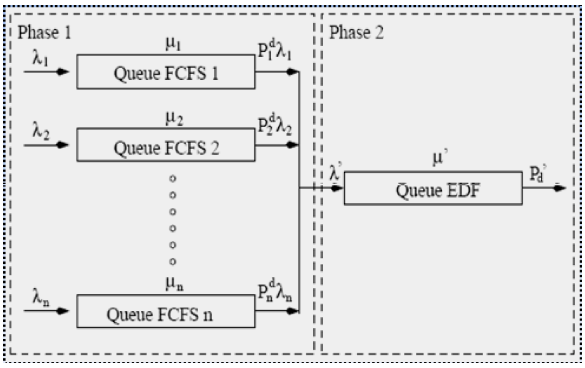

Fig 1: Analytical Model for the Two-phase LAQ Algorithm

Unknown Input for Phase 2: The input for Phase 2 is the data not served in Phase 1, which is an unknown parameter to be derived in analysis.

In this work, an elegant queuing model as shown in Fig. 1 is constructed to describe the two-phase LAQ algorithm. Our key idea is to creatively use the M/D/1+D (for Markov arrival, deterministic service time, one server, plus deterministic delay budget) queuing model with FCFS and EDF queuing disciplines. We consider a general scenario, where *n* rtps and/or nrtps connections are established. Connection *i* has an arrival rate of 8_i a delay budget of J_i and a minimum reserved bandwidth of MRR_i. For the sake of analytic tractability, we assume that the data arrival forms a Poisson process and all queues have infinite size. Other types of traffic (such as the more practical burst traffic) are studied through simulations.

Following the LAQ scheduling algorithm, the queuing model also consists of two phases. In Phase 1, since the connections are served with their reserved bandwidth (i.e. MRR_i), each of them can be modeled as a separate and independent queue. Assuming the data packet size is fixed, the service rate is a constant. More specifically, the service rate of Queue i (for Connection i) is $m = \frac{MRR_i}{L}$ (1). Where L denotes the length of the data packet. Clearly, the service rate of Queue *i* is proportional to the MRR of Connection *i*. Note that, data packets that cannot be served in Phase 1 due to the limited MRR will all be passed to Phase 2. Therefore, we artificially set a delay bound of T (which is the period of one frame) for the queues in Phase I, in order to track the "dropped" data packets of Phase 1, which will be the inputs for Phase 2.

Based on the above discussion, we arrive at an M/D/1+D queue for each connection in Phase 1, with an arrival rate of ¸*i*, a service rate of ¹*i* and a delay budget of

T. The M/D/1+D queue under FCFS queuing discipline has been well studied. According to [14], the packet drop rate of Queue *i* is:

$$P_i^d = 1 - \frac{1}{r_i} + [r_i + r_i^2 e^{l_i t} \sum_j^r = 0(-1)\, j \frac{\left(l_{it - r_i}\right)^i}{j!} e^{-r_i^j}]^{-1} \quad (2)$$

Where D_i equals $\frac{l_i}{m_i}, t$ is the delay bound which is set to *T* and *r* is the integer satisfying $(m_i t - 1) < r < m_i t$

The packets "dropped" from Phase 1 become the input of Phase 2. Because such "packet dropping" is random, the "dropped" packets of Queue i also form a Poisson process, with mean arrival rate of $l_i \times P_i^d$

As shown in Fig. 1 the input of the queue is the aggregation of the "dropped" packets from all queues in Phase 1. Since the "dropped" packets from each queue are Poisson, the aggregation of them is also Poisson, with a mean arrival rate of $l' = \sum_{i=1}^n P_i^d \times l_i$ (3)

The service rate in Phase 2 denoted by μ_i is estimated as follows. The amount of bandwidth available for Phase 2 is what has been left after Phase 1.

We assume that CAC ensures the total reserved bandwidth never greater the total available bandwidth, i.e. $\sum_{i=1}^n MRRi \le W$, where *W* denotes the total available bandwidth. Thus the remaining bandwidth comes from:
~the bandwidth that is not reserved by any connections, i.e. $W - \sum_{i=1}^n MRRi$ and

C The bandwidth that is reserved but not actually utilized in full by some connections.

When a connection is idle, its reserved bandwidth is not used and thus contributing to the bandwidth of Phase 2.

For Queue i in Phase 1, the fraction of time when packets are served is:

$$\frac{t_{busy}}{t} = \frac{l\left(1 - P_i^d\right) \times t \times \frac{1}{m_i}}{i} = l\left(1 - P_i^d\right) \times \frac{1}{m_i} \quad (4)$$

Where t denotes any given time interval and t_{busy} represents the busy period within t. In the above equation, $l\left(1 - P_i^d\right) \times t$ is the actual number of arrivals in *t* (excluding those being dropped) and $\frac{1}{m_i}$ is the service time for each packet. Thus the fraction of idle time, denoted by P_i^0, is $P_i^0 = 1 - l_i\left(1 - P_i^d\right) \times \frac{1}{m_i}$ (5)

Accordingly, the overall service rate in Phase 2 is

$$m' = \frac{W - \sum_{i=1}^n MRRi}{1} + \sum_{i=1}^n P_i^0 m_i \quad (6)$$

Actually, Equation (4) is a form of Little's theorem [13] and 1-P^0 is also known as the utilization factor. We model Phase 2 as an M/D/1+D queue, with an arrival rate of δ' and a service rate of μ'. In addition, each packet in the queue is associated with a delay budget, e.g. J_i for the packet from Connection i. The packet drop rate P_d' obtained above is with respect to the input of Phase 2, i.e. the arrivals with a mean rate of δ'. As shown in Fig 1, the actual inputs of the entire system are the arrivals with means of $\delta_1, \delta_2, ...\delta_n$ in Phase 1. Therefore, the overall dropping probability is:

$$P_d = P_d' \frac{l'}{\sum_{i=1}^n l_i} \quad (7)$$

And the throughput is 1-P_d

Simulations and Discussion: We have carried out extensive simulations to verify the correctness of the proposed analytical model for the LAQ algorithm and to compare its performance with other scheduling algorithms, under a broad set of simulation parameters. In particular, we consider several comparable scheduling algorithms, including WRR, EDF and DFPQ [7] (which is a representative WiMAX scheduling algorithm and has been patented and well received).

Besides packet drop rate and throughput that have been studied in analysis, we are also interested in the fairness performance, which is measured by Jain's Fairness Index [15] defined as follows:

$$f\left(X_1, X_1, ...X_n\right) = \frac{\left(\sum_{i=1}^n Xi\right)^2}{n\sum_{i=1}^n X_i^2} \quad (8)$$

Where x_i is the normalized throughput of connection i and *n* is the total number of connections. Here we use the normalized throughput of a connection, i.e. $X_i = \frac{\mathrm{Th_i}}{\mathrm{MRR_i}}$, with Thi and MRRi stand for the connection i's actual data rate and reserved data rate, respectively. The Jain's Fairness Index ranges between 0 and 1. The higher the index, the better the fairness. If *Thi = MRRi* for all *i*, or in other words, every connection obtains its reserved data rate, then *xi* = 1 for all *i* and Jain's Fairness Index equals 1. All simulations and analytic calculations are done using NS2 (2.34) simulator.

Table 1: Raw Data Rate.

SS ID	SS1	SS2	SS3	SS4
Modulation type	BPSK	QPSK	16QAM	64QAM
Inner code rate	2-Jan	3-Feb	4-Mar	6-May
Bits / symbol	1	2	4	6
Raw data rate Rb(Mbps)	3.716	9.9094	22.2962	37.1603
Bytes / mini slot	1.875	4.999	11.247	18.745

Table 2: MRR and Delay Budget of Different connections.

rtps	rtps1	rtps2	rtps3	rtps4
MRR (kbps)	19.2	64	384	1024
Delay (ms)	10	30	20	40
nrtps	nrtps1	nrtps2	nrtps3	nrtps4
MRR (kbps)	24	48	256	768
Delay (ms)	100	130	170	200

Table 3: Distribution of the 8 Connections in 4 SS.

SS ID	SS1	SS2	SS3	SS4
rtps1+ nrtps1	**s**			
rtps2 + nrtps2	**s**	**s**	**s**	
rtps3 + nrtps3		**s**	**s**	**s**
rtps4 + nrtps4				**s**

The simulation parameters are bring in following tables:

Each SS establishes a number of connections to the BS in our simulation. We consider four rtps connections, named rtps1, rtps2, rtps3 and rtps4 and four nrtps connections, named nrtps1, nrtps2, nrtps3 and nrtps4. As shown in Table 2, each type of connection is associated with an MRR and a delay budget.

Every SS has four connections as shown in Table 3. For example, SS1 has these four connections: rtps1, nrtps1, rtps2 and nrtps2. As a result, there are 16 connections, which request a total MRR of 537 mini slots. We consider two types of traffic in our simulations, i.e. the popular Poisson traffic and the burst traffic generated by the Markov Modulated Poisson Process (MMPP) model, as discussed below.

Poisson Traffic: First, we use Poisson traffic, with packet size of l = 800 bytes. The arrival rates of all connections except rtps3 are fixed and match their reserved data rates. For example, the packet arrival rate of Connection i equals MRRi=l. Meanwhile, we vary the arrival rate of rtps3, in order to study the influence on the scheduling results. We have obtained analytic and simulation results for LAQ, as well as the simple WRR and EDF schemes. For DFPQ approach in [7], the results are obtained via simulation only.The analysis and simulations are compared in Fig. 2, where X-axis indicates the traffic load of rtps3 (where D equals arrival rate over $MRRi=l$) and Y-axis shows the overall packet drop rate of all connections. The simulation result of DFPQ algorithm is also depicted in Fig. 2. We observe that EDF, LAQ and DFPQ have similar packet drop rate, while the drop rate of WRR is much higher, because WRR does not consider the delay budget and is more likely to drop real time data packets. Note that, with the increase of the traffic load of rtps3, the total traffic load becomes significantly higher than the total available bandwidth. Therefore, all four algorithms exhibit high drop rate. Simulation results of the Jain's Fairness Index are shown in Fig. 3. As we can see, the fairness index of EDF drops dramatically when the traffic load of rtps3 becomes higher than its MRR, because the additional real time traffic (with tight delay budget) aggressively takes bandwidth from the nrtps connections under the EDF algorithm and thus leading to low throughput and unfairness to non-real time traffic. When D of connection rtps3 is greater than 1, the fairness index of both LAQ and DFPQ drops, but DFPQ's fairness index drops more rapidly, exhibiting worse fairness performance than LAQ. The reason can be two-folded. First, DFPQ always gives real time traffic higher priority than non-real time traffic, resulting in unfairness. The other reason is that DFPQ does not guarantee the minimum reserved rate of each real time connection since EDF is deployed within all the real time connections. Among the four scheduling algorithms, WRR's fairness index is the stablest when the input traffic varies. In summary, the fairness index of LAQ is better than that of DFPQ and EDF and comparable to WRR.

Burst Traffic: To study the performances of the four scheduling algorithms under burst traffic, we use Markov Modulated Poisson Process (MMPP) model to generate data packets.

The burstiness of the traffic is defined as $b = \frac{l_{\max}}{l_{avg}}$.

The higher the value of b, the more bursty the traffic is. When b equals to 1, the MMPP model is equivalent to the Poisson model.

To focus on the impact of burst traffic, we vary b and fix the arrival rate of all connections, i.e. let 8_i= 1/5MRRi/L for rtps3 connections and 8_i= MRRi/L for other connection. The packet drop rates of the four scheduling algorithms under burst traffic are shown in Fig. 4. As can be seen, WRR is rather vulnerable to the burst traffic.

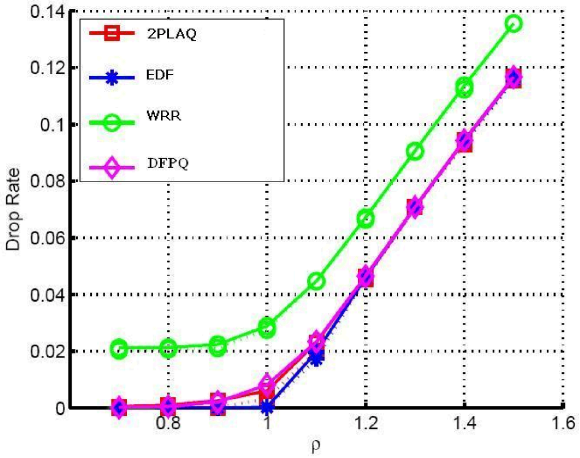

Fig. 2: Drop Rate under Poisson Traffic.

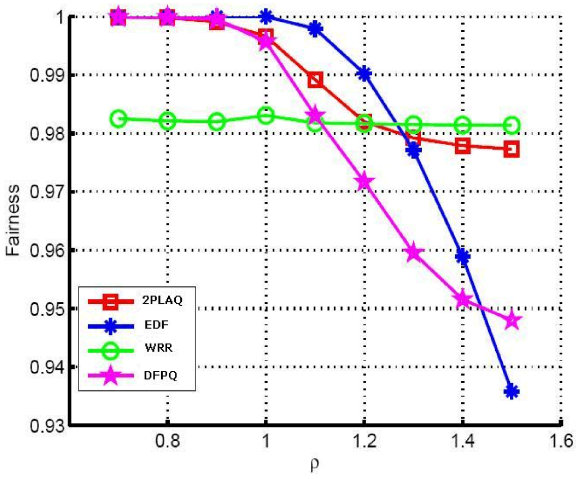

Fig. 3: Fairness under Poisson Traffic

With the increase of *b*, its drop rate increases dramatically. On the other hand, EDF, LAQ and DFPQ can maintain a reasonably low packet drop rate even when b is large. The fairness of the scheduling algorithms under burst traffic is shown in Fig.4. As we can see, LAQ always maintains a high fairness index, while the fairness of EDF algorithm is the worst among the four algorithms.

Notice that when *b* increases, the fairness index of EDF becomes higher. This is due to the fact that some real time packets of rtps3 connection are dropped under high burstiness and thus the throughput of rtps3 decreases to a value more closer to its reserved throughput. Accordingly, the normalized throughput of rtps3 becomes close to 1, which appears more fair. On the contrary, the fairness of WRR drops with the increase of traffic burstiness.

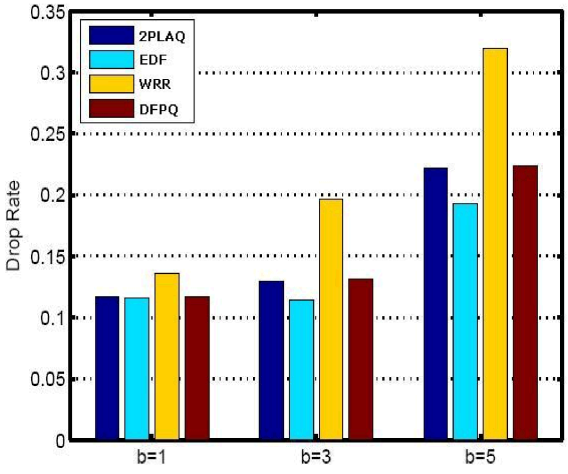

Fig. 4: Drop rate under Burst Traffic.

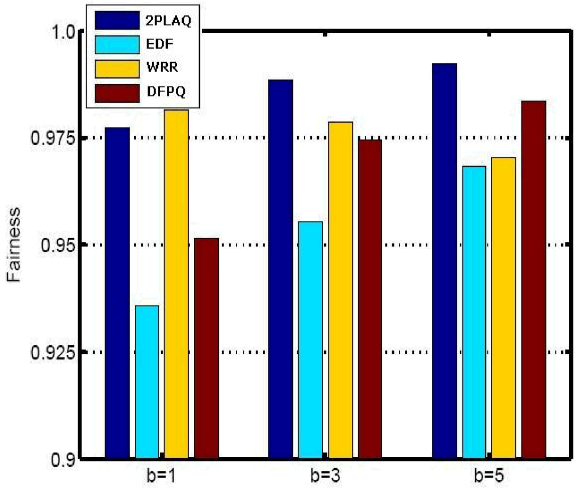

Fig. 5: Fairness under Burst Traffic.

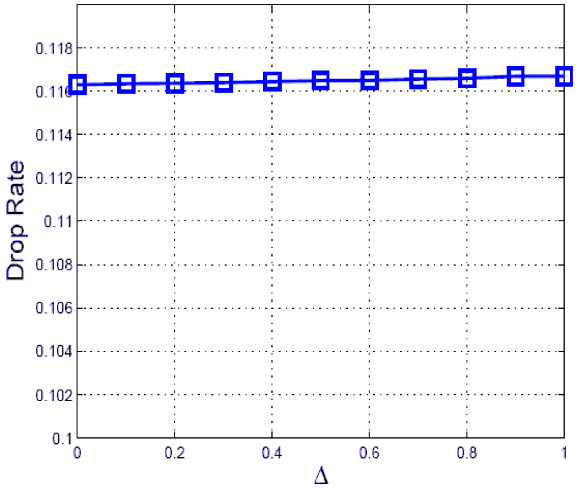

Fig. 6: Drop rate-Impact of Bandwidth Allocation between Two Phases ( )).

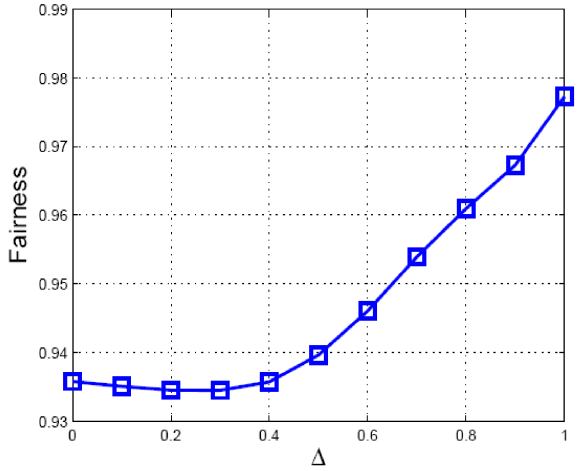

Fig. 7: Fairness-Impact of Bandwidth Allocation between Two Phases ( )).

In general, there is a trade-off between the drop rate and fairness when) varies from 0 to 1. For example, a small) indicates more bandwidth allocated to Phase 2, which results in a smaller drop rate but higher unfairness. However, our simulation results show that the algorithm Achieves higher overall performance under larger). As can be seen in Fig. 6 and 7, when) increases from 0 to 1, the drop rate increases only marginally by about 0.1%, while the Jain's Fairness Index increases from around 0.93 to 0.98, which is close to the perfect fairness. This result justifies our choice of letting) = 1 in LAQ.

CONCLUSION

In this article, we have presented a two-phase Learning Automaton Queuing (LAQ) algorithm for uplink scheduling in the WiMAX network. It aims to strike the balance between fair bandwidth allocation and delay requirement. In order to gain deep understanding of and insights into LAQ algorithm, we have established an elegant queuing model to derive in theory the performance metrics in terms of packet drop rate and throughput. The analytic model has been verified by extensive simulations, carried out on the basis of a broad set of parameters according to WiMAX physical layer standards. In simulations, we have considered both popular Poisson traffic and practical burst traffic that is modeled by the Markov Modulated Poisson Process (MMPP). The simulation results have verified the correctness of our analytical models. We have also compared LAQ with WRR, EDF and Deficit Fair Priority Queue (DFPQ). Both analytic and simulation results have clearly shown that LAQ algorithm effectively achieves low packet drop rate and high throughput while maintaining the fairness among different connections.

REFERENCES

1. IEEE 802.16-2004, 2004. IEEE Standard for local and metropolitan area Networks, Air Interface for Fixed Broadband Wireless Access Systems,
2. "IEEE 802.16e, 2004. IEEE Standard for local and metropolitan area networks, Air Interface for Fixed Broadband Wireless Access System, Amendment",
3. Hawa, M., 2003. "Stochastic Evaluation of Fair Scheduling with Applications to Quality-of-Service in Broadband Wireless Access Networks", Ph.D. dissertation, University of Kansas.
4. Wongthavarawat, K. and A. Ganz, 2003. "IEEE 802.16 Based Last Mile Broadband Wireless Military Networks with Quality of Service Support" in Proc. Mil. Commun. Conf, pp: 779-784.
5. Liu, Q., X. Wang and G. Giannakis, 2006. "A Cross-Layer Scheduling Algorithm with QoS Support in Wireless Networks," IEEE Trans. Veh. Tech., 55(3): 839-847.
6. Niyato, D. and E. Hossain, 2006. "Queue-aware Uplink Banwidth Allocation and Rate Control for Polling Service in IEEE 802.16 Broadband Wireless Networks," IEEE Trans. Mobile Comp., 5(8): 668-679.
7. Chen, J., W. Jiao and H. Wang, 2005. "A Service Flow Management Strategy for 802.16 Broadband Wireless Access System in TDD Mode," in Proc. IEEE ICC, pp: 3422-3426.
8. Raghu, K.R., S.K. Bose and M. Ma, 2007. "Queue Based Scheduling for IEEE 802.16 Wireless Broadband," in Proc. 6th IEEE Int. Conf. ICICS, pp: 1-5. 2007.
9. Sharma, V. and N. Vamaney, 2007. "The Uniformly-Fair Deficit Round-Robin (UF-DRR) Scheduler for Improved QoS Guarantees in IEEE 802.16 WiMAX Networks," in Proc. Mil. Commun. Conf, pp: 1-7.
10. Shejwal, A. and A. Parhar, 2007. "Service Criticality Based Scheduling for IEEE 802.16 WirelessMAN," in Proc. 2nd IEEE Int. Conf. AusWireless, pp: 12-18.
11. Salodjar, N. and A. Karandikar, 2008. "An Indexing Scheduler for Delay Constrained Scheduling with Applications to IEEE 802.16," in Proc. IEEE WCNC, pp: 1471-1476.

12. Bai, X., A. Shami and Y. Ye, 2008. “Robust QoS Control for Single Carrier PMP Mode IEEE 802.16 Systems,” IEEE Trans. Mobile Comp., 7(4): 416-429.
13. Kleinrock, L., 1979. Queueing Systems, Volume 1: Theory. Hoboken, NJ: John Wiley and Sons,
14. Kok, A.G.D. and H.C. Tijms, 1985. “A Queueing System with Impatient Customers,” J. Appl. Prob., 22(3): 688-696.
15. Jain, D.M.C.R. and W. Hawe, 1984. “A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems”, dEC Research Report, TR-301.
16. Hall, J. and P. Mars, 1998. Satisfying QoS with Learning Based Scheduling Algorithm. In 6th International Workshop on Quality of Service, pp: 171-176.