

A Prioritized Load Aware Weighted Round Robin (PLAWRR) algorithm in Broadband Wireless Networks

Ibrahim S., Abubakar R., Nasir A.S., and A.U. Moyi.

Abstract— Broadband Wireless networks (BWNs) provide a reliable internet access for the delivery of high-speed multimedia applications. The BWNs such as WiMAX provides quality of service (QoS) support for heterogeneous service classes with diverse QoS requirements. Scheduling algorithm is one of the mechanisms used to assure QoS. The existing scheduling algorithm uses a priority value to prioritize traffics according to varying traffic conditions. However, it wastes network resources due to failure to consider channel conditions; thus, lead to increase in delay and packet loss as well as decrease in throughput. In this paper, a prioritized load aware weighted round robin (PLAWRR) algorithm is proposed to improve resource utilization. The PLAWRR algorithm employs a modified value priority according to not only traffic load but also channel condition and throughput history for traffics prioritization. It also introduces a new dynamic weight according to the prioritization value. The performance of the proposed algorithm is evaluated using simulations. The results show that the proposed PLAWRR achieves superior performance compared to the existing algorithm in terms of delay and packet loss as well as increase in throughput.

Index Terms—Quality of services; Wireless Metropolitan Network; WiMAX; and Scheduling algorithm .

I. INTRODUCTION

Broadband Wireless Access (BWA) has gained growing acceptance worldwide recently [1]. Due to the increasing demand for wireless multimedia applications such as Voice over Internet Protocol (VoIP), Internet Protocol Television (IPTV), interactive gaming and multimedia conferencing, to efficiently manage bandwidth becomes a challenge. Several Broadband Wireless Access (BWA) technologies are used in wireless networks such as Long Time Evolution (LTE) and IEEE 802.16 standard. The IEEE 802.16 standard also known Worldwide Interoperability for Microwave Access (WiMAX) had several advantages including ease and low cost of deployment, first-mile/last-mile access, and quality-of-service (QoS) support for multimedia applications [2]. The WiMAX addresses broadband access technology for Wireless Metropolitan Area Networks (WMANs). It is anticipated to be a viable alternative to traditional wired broadband techniques due to its cost-competitiveness, ubiquitous access, and Quality of Service (QoS) capabilities. Because it is a wireless technology, in which resources are limited such as bandwidth; how to manage these resources while maintaining QoS to diverse applications become a critical issue. Therefore, efficient scheduling algorithm is essential in this network.

Several scheduling algorithms have been proposed to improve resource utilization and guarantee QoS to diverse applications [2-4]. In [2], a Load Aware Weighted Round Robin (LAWR) algorithm is proposed to improve network efficiency. The LAWR employs dynamic weight in order to reduce not only delay and packet loss but also improve throughput. However, it increases delay of real time traffic. In [3], a Low Latency-Weighted Round Robin (LL-WRR) is proposed to improve latency and fairness. The algorithm uses the coefficient γ in order to improve the performance of the network but the computation of γ introduces additional complexity. In [4], a priority load aware scheduling (PLAS) algorithm is proposed to minimize delay in real time traffics. The PLAS algorithm uses a priority value to prioritize real time traffics over non-real time traffics. The algorithm reduces the average delay but lead to inefficient resource utilization and hence poor QoS because channel condition is ignored.

In this paper, a channel -aware weighted round robin (PLAWRR), for BWAs in the downlink direction to improve upon the efficiency of PLAS. The PLAWRR algorithm employs a modified prioritization value according to the traffic characteristics and channel conditions as well throughput history to prioritize traffics. It also introduces a new dynamic weight based on a priority value to increase number of packets to be served. The performance of the proposed algorithm is evaluated using discrete-event simulations. The simulation results show that the performance of the proposed algorithm outperforms the compared algorithm in terms of delay and packet loss as well as throughput.

This paper is organized as follows: Section II provides related works on scheduling algorithms. In Section III, describes the proposed algorithm. Section IV presents the performance evaluations, and summarizes in Section V.

II. RELATED WORKS ON SCHEDULING ALGORITHMS

This section presents the various scheduling algorithms adopted and proposed in WiMAX networks. The way the algorithms operate, strengths and weaknesses are discussed as follows:

Round Robin (RR) algorithm [5] is a simple approach designed for time-sharing systems ,which considers priority on queues. RR starts by classifying packets into low- and

I.S. is with the Department of Information and Communication Technology, Usmanu Danfodiyo University Sokoto (UDUS), Nigeria (e-mail: Ibrahim.saidu@udusok.edu.ng).

A. R. is with the Department of Computer Science, UDUS, Nigeria (e-mail: abroko@yahoo.com).

N.A.S. is with the Directorate of ICT, Umar Musa Yar'adua University, Katsina, Nigeria (e-mail: nashinkafi@gmail.com).

A.M. is with the Department of Mathematical Sciences, Federal University Gusau, , Nigeria (e-mail: amoyi@gmail.com).

high-priority service classes. Once mapped, these packets are assigned to dedicated queues. Each nonempty queue is then allowed to send a packet, starting with the highest priority class, without any further priority assignment by the scheduler. RR serves all non empty queues in cyclical order; once the queue is served from a given class in the current service round, it is not served again until all unserved queues have been served within a single round. If there are still nonempty queues, the algorithm moves to a next round until they are emptied. RR is therefore fair when equal packet lengths are used but unfair when variable packet lengths are used by different queues. Moreover, it cannot guarantee different QoS requirements for different service classes [6].

Weighted Round Robin (WRR) is a modification of RR proposed by [7] that considers the QoS requirements for each class. Static weights are used to distinguish the QoS requirements of each class and also to determine its required bandwidth. Each class has a list of queues, and each of these has an associated weight counter. If a queue is picked for service, its (static) weight is assigned to its counter, which specifies the number of packets to be transmitted in a round. The weight of each queue is calculated by considering the QoS requirements of each class. The various QoS parameters used are traffic priority, MSTR, and MRTR [8]. The algorithm serves each queue in RR fashion from highest to lowest priority class. Packets from each queue are served in each round from those whose weights are nonnegative. If a packet is sent, a queue weight counter is decremented. Packets continue to be sent until the counter reaches zero and the queues in all classes are empty. Finally, all queue counters are reset to their weight values. Therefore, WRR is fair when equal weights and equal packets are used by all the queues in each class. However, the use of static weighting to serve classes with variable priority levels under input burst traffic may lead to rapid growth of their queues and, consequently, to further packet delay as well as packet loss. It is also unfair to queues with variable packet lengths. Furthermore, unlike real-time queues, which are given higher weights, non-real-time queues are given lower weights [8], thus imposing a delay on lower priority classes.

Modified Weighted Round Robin (MWRR) is a variation proposed by [9] to overcome this penalty for lower priority classes. It starts by computing the WRR weight for each queue based on priority and the number of all nonempty queues. This weighting permits each queue to transmit a certain number of packets in a single service round. Thus, the total number of packets a WRR scheduler can deliver per service round is fixed. The MWRR scheduler multiplies each WRR weight counter by a constant integer value in order to increase the service round. This algorithm reduces the average delay and increases average throughput, particularly for lower priority classes, by lengthening the size of the service round over WRR. However, the multiplier used is static, which may lead to either increase in delay or decrease in throughput when it is inappropriately chosen.

Adaptive Weighted Round Robin (AWRR) is yet another variant, proposed by [10], again to avoid the problem of starvation of lower priority classes. It uses two schedulers: An input scheduler aims to prioritize video streams with superior quality (HD and SD) and consists of a High-Priority (HP) buffer for, e.g., UGS, rtPS, and nrtPS, and a Low-priority Buffer (LB), which includes rtPS-web-TV, rtPS-mobile-TV, nrtPS, and BE. On the other hand, the objective of the output

scheduler is to regulate data flow and manage all the service classes. Both schedulers use AWRR to adjust the weighting of the service classes. A threshold value is set for each class, which triggers dynamic weight adjustment whenever a threshold is exceeded. The HP queues have lower thresholds than do the LP queues. The input scheduler controls the weights of the HP and LP queues based on the HP traffic load and buffer size. Control of the weights from the queue allows the algorithm to achieve minimum throughput of BE traffic under conditions of network stress and also to give preferential treatment to HP traffic. The dynamic weight is calculated using an algorithm in which the weights need to be positive to ensure minimum throughput for LP traffic, and an arbitrary constant is used to favor HP traffic classes such as UGS and rtPS. However, it employs a complex calculation to compute the weights and applied in WiMAX multihop networks.

In [2], a load-aware weighted round robin(LAWRR) algorithm is proposed to improve network efficiency under bursty traffic condition. The LAWRR introduces a dynamic weight which determine the weight of each queue in the various classes according to a traffic condition and the WRR static weight. The algorithm not only reduces delay and packet loss but also increases throughput. However, it increases delay of real time traffics because of failure to prioritize real time traffic over non-real time traffic.

Low Latency-Weighted Round Robin (LL-WRR) is another variant, proposed by [3], that enhances the latency and fairness of real time services when the number of connections increases. The algorithm introduces a coefficient (γ), instead of a constant integer such as is used in [9], which is a function of the number of connections present in the network. The coefficient γ is computed at the beginning of the WRR cycle and multiplied by the WRR of each connection. The value of γ varies as the number of connections increases. To keep the latency low, γ is decreased with an increase in the number of connections, because the latency is proportional to γ . The algorithm also improves fairness when the value of γ reduces faster as the growth rate of the connections, then both the number of packets to be transmitted in a round and the weighted fairness index decrease. The algorithm is able to achieve low latency and improve fairness. However, the computation of γ introduces additional complexity.

In [4], a priority load aware scheduling (PLAS) algorithm is also proposed to decrease delay in real time traffics. The PLAS algorithm employs a priority value to prioritize real time traffics over non-real time traffics. The algorithm minimises the average delay but waste resources due to failure to consider channel condition.

To date, none of the algorithms mentioned proved capable of further reducing delays and packet loss while improving the throughput in IEEE 802.16 networks by considering not only traffic load in each queue but also channel conditions and throughput history for the prioritization of real time traffic.

III. PROPOSED PLAWRR

This section presents the proposed PLAWRR algorithm to provide efficient resource utilization. The algorithm firstly utilizes a static weight derived as follows:

$$Sw_i = \frac{MRTR}{\sum_{i=1}^n MRTR_i} \quad (1)$$

Where MRTR is the minimum reserved traffic rate of queue i, n is the total number of calls in a class.

Secondly, the dynamic coefficient of queue i at round r is derived as

$$D_{i,r} = \left\lceil \frac{L_{i,r}}{R_r+1} \right\rceil \quad (2)$$

Where $L_{i,r}$ is the load of queue i at round r and R_r is the root mean square error (RMSE) computed as follows:

$$R_r = \sqrt{V_r} \quad (3)$$

Where V_r is the load variance of the queues at the current round

Thirdly, the dynamic weight is derived as

$$Dw_{i,r} = D_{i,r}Sw_i \quad (4)$$

The Equation (4) is employed to determine the number of packets to be served from each queue.

Fourthly, a modified weight value is introduced to improve resource utilization. The value is computed not only based on traffic loads but also channel conditions and throughput history adopted from [11]. It schedules users with the highest priority as follows

$$P_k(t) = \begin{cases} \frac{CC_i(t)}{1+R_i(t) * T_i(t)/N_i}, & \text{if real traffic} \\ \frac{CC_i}{1+T_i}, & \text{if non real time traffic} \end{cases} \quad (5)$$

Where $R_i(t)$ is the real-time service minimum rate requirement, $CC_i(t)$ is the channel condition at time t, $T_i(t)$ is the historical throughput of that class of connection, N_i is the number of connections at the ith queue.

Finally, a new dynamic weight is obtained as

$$Mw_{i,r} = P_k(t) \cdot Dw_{i,r} \quad (6)$$

The results from Equation (6) is the weight of each queue which determines the number of packets to be served in each queue. Whenever a packet is scheduled from a queue, the weight counter of the queue decrements by 1 and the queues are served in a round robin fashion, scheduling packets from each queue. Whenever the weight counter of each queue is zero or all queues are empty, the scheduler resets the counter.

IV. PERFORMANCE EVALUATION

This section presents evaluation of the proposed algorithm compared to the PLAS algorithm using simulation. The simulation parameters and topology are adopted from [3] as shown in Table 1 and Figure 1, respectively. The topology consists of one Base Station (BS), 35 Mobile Stations (SS) distributed around the BS, and an application server. The server generates two types of traffics: real time (UGS,rtPS, and ertPS) and non real time (FTP and BE) with the assumption that each user traffic is carried by one SS and that each user can only utilize one type of traffic at a time instance. The performance metrics that are used are average delay, packet loss and throughput as derived.

Parameter	Value
BS frequency	2.5 GHz
Duplexing mode	TDD
System bandwidth	5 Mbps
DL/UL ratio	2:1(29:18 OFDM symbol)
Frame length	5 ms
Cyclic prefix duration	11.43 μs
Basic symbol	91.43 μs
FFT	1,024
PHY OFDMA	OFDMA
DL permutation zone	PUSC
MAC PDU length	Variable
Fragmentation	Enable
ARQ and packing	Disable
DL-UL MAPS	Variable

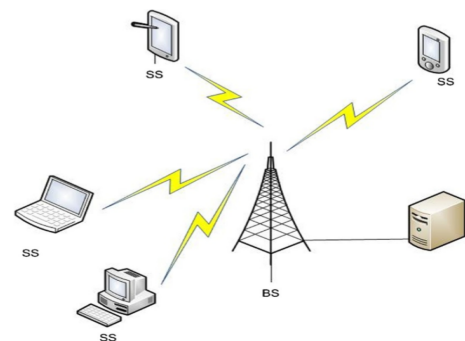


Figure 1: Simulation network topology

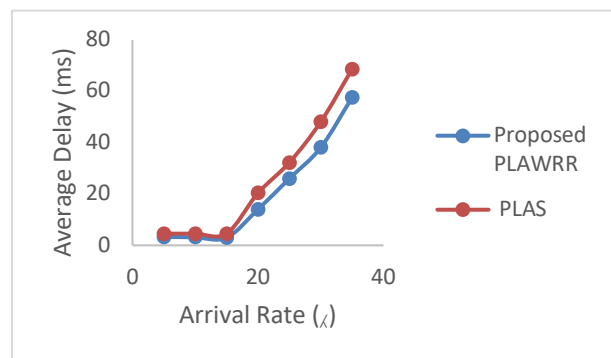


Figure 2 Average Delay per MS

Figure 2 shows the average delay per MS for the Proposed PLAWRR and PLAS. The figure illustrates that the PLAWRR performs better than the PLAS because it considers the channel condition and the historic throughput to minimise the waste of resources. The resources obtained help in increasing the number of packets to be transmitted as well as reduces the retransmission of dropped or corrupted packets.

Table 2 Simulation parameters in [11]

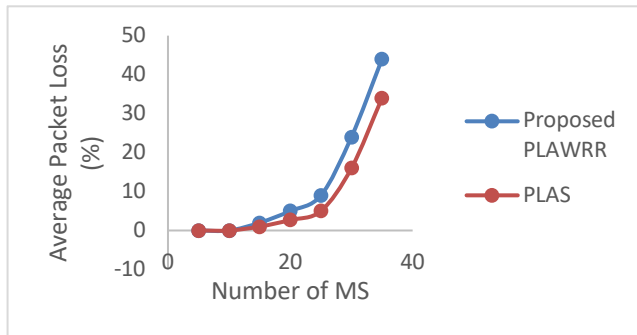


Figure 3 Average Packet Loss per MS

Figure 3 shows average packet loss per MS for the two algorithms (Proposed PLAWRR and PLAS). The proposed PLAWRR achieves superior performance because of consideration to channel conditions as well as historic throughput to increase the resource availability in order to avoid prolong queueing, which may lead to packet loss when resources are inefficient.

V. CONCLUSION

In this paper, a PLAWRR is proposed to improve resource utilization. The algorithm introduces a prioritization value according to traffic loads, channel conditions and throughput history in order to prioritize traffics. It also employs a new dynamic weight according to the priority value to increase the number of packets to be served in a queue. The proposed algorithm is evaluated through extensive simulation. The results illustrate that the proposed algorithm achieves a superior performance compared to PLAS in terms of average delay, packet loss and throughput.

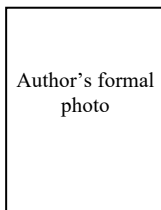
ACKNOWLEDGMENT

I would like to thank the Faculty of Engineering and Environmental Design, UDUS, Nigeria for financial support.

REFERENCES

- [1] I. Saidu, "Quality of Service Management Algorithms in WiMAX Networks", PhD thesis Department of Electrical and Computer Engineering University Putra Malaysia, September 2015.
- [2] I. Saidu, S. Subramaniam, A. Jaafar, and Z.A., Zukarnain, "A load-aware weighted round-robin algorithm for IEEE 802.16 networks." *EURASIP Journal on Wireless Communications and Networking*, 2014(1), p.226.
- [3] J.Z. Patel and U. Dalal, "Design and Implementation of Low Latency Weighted Round Robin (LL-WRR) Scheduling for High Speed Networks," *International Journal of Wireless & Mobile Networks*, vol. 6, no. 4, pp. 59–71, 2014.
- [4] Mohammed, A., Saidu, I. and Abdulazeez, A., 2017, December. A Priority Load-Aware Scheduling Algorithm for Wireless Broadband Networks. In *International Conference on e-Infrastructure and e-Services for Developing Countries* (pp. 49-59). Springer, Cham.
- [5] E. L. Hahne and R. G. Gallager, "Round robin scheduling for fair flow control in data communication networks," NASA STI/Recon, Tech. Rep. 30047, 1986.
- [6] S. Mahasweta and S. Harpreet, "A QoS aware packet scheduling scheme for WiMAX", *Proceedings of IAENG Conference on World Congress on Engineering and Computer Science*, Berkeley, 1 Berkeley, California, USA, 2009.
- [7] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis, "Weighted round robin cell multiplexing in a general-purpose ATM switch chip," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 8, pp. 1265–1279, 1991.

- [8] N. A. Ali, P. Dhrona, and H. Hassanein, "A performance study of uplink scheduling algorithms in point-to-multipoint WiMAX networks," *Computer Communications*, vol. 32, no. 3, pp. 511–521, 2009.
- [9] W. Mardini and M. A. Alfool, "Modified WRR scheduling algorithm for WiMAX networks," *Network Protocols and Algorithms*, vol.3, no.2, pp. 24–53, 2011.
- [10] M.-e.-A. Brahmia, A. Abouaissa, and P. Lorenz, "Adaptive scheduling mechanism for IPTV over WiMAX IEEE 802.16 j networks," *International Journal of Communication Systems*, vol.27, no.7, pp.1009–1019, 2014.
- [11] Nie, Wei, Houjun Wang, and Jong Hyuk Park. "Packet scheduling with QoS and fairness for downlink traffic in WiMAX Networks." *Journal of Information Processing Systems* 7, no. 2 (2011): 261-270.



Ibrahim Saidu obtained a BSc. Mathematics and an MSc. Mathematics from Usmanu Danfodiyo University Sokoto in 1998 and Bayero University Kano in 2009, respectively. He also obtained additional Bachelor in Information and Communication Technology from Al-Madinah International University, Malaysia and Masters in Computer Science with specialization in Distributed Computing from University Putra Malaysia (UPM) in 2010. In addition, he obtained

PhD in Computer Networks from UPM in 2015 and attended several ICT related short courses from Rice University, University of Washington, Vanderbilt University, University of New-Mexico, Hasso Plattner Institute Germany, Telekoms Academy, UK and ITU Academy.

He is currently working with the Department of Information and Communication Technology, UDUS Nigeria. He has published several articles from International refereed Journals and Conferences. His area of interests are Resource Management, QoS, and Energy Saving in Wireless Networks.

Dr. Ibrahim Saidu is a member of INSTICC, CPN, NCS, and AIT