Scalable Scheduling with Burst Mapping in IEEE 802.16e (Mobile) WiMAX Networks

Mukakanya Abel Muwumba and Idris A. Rai

Makerere University
P.O. Box 7062
Kampala, Uganda
abelmuk@gmail.com
rai@cit.mak.ac.ug

Abstract. **O**ne **C**olumn **S**tripping with non-increasing **A**rea first mapping algorithm (OCSA) was proposed by Chakchai So-In et al to schedule bursts on downlinks of base station by giving priority to the largest bursts to OFDMA frame. However, size-based scheduling that favor large items are known to exhibit poor average delay performance especially under workload distributions that are highly skewed (i.e., heavy tailed workloads). In this paper, we first study OCSA and use numerical results to show that it starves short bursts at high loads. We then propose improvement to OCSA (iOCSA) and a new algorithm called **O**ne **C**olumn **S**tripping with **I**ncreasing **A**rea first mapping algorithm (OCSIA). In contrast to OCSA, OCSIA gives priority to short bursts. Our detailed numerical results to compare OCSIA to OCSA under varying workload distributions clearly show iOCSA improves the performance of OCSA, and OCSIA significantly outperforms OCSA under heavy tailed workloads without starving large bursts.

Keywords: bursts, heavy-tailed, scheduling and workloads.

1 Introduction

Scheduling plays a vital role in the QoS provision therefore design of efficient and scalable scheduling algorithms for communication systems is very important. WiMAX or IEEE 802.16e however doesnt specify scheduling leading to concerted efforts by researchers to propose a number of potential scheduling schemes for WiMAX. As such various scheduling algorithms have been proposed for WiMAX [1,2,3,4,5,6] and more other exist. The work in this paper is closely related to bursts *size-based* scheduling schemes that were proposed for mobile WiMAX downlink subframe such as OCSA and eOCSA [1,6].

OCSA is a two dimensional rectangular mapping of bursts on the Orthogonal Frequency Division Multiple Access (OFDMA) downlink subframe in IEEE 802.16e mobile WiMAX. OCSA optimizes frame utilization and maximizes the bursts allocation by giving priority to the largest bursts. However, it is a known fact that scheduling policies that favor large bursts perform very poorly under

R. Popescu-Zeletin et al. (Eds.) AFRICOMM 2011, LNICST 92, pp. 84–95, 2012.

heavy-tailed workload distributions [7]. Such workload distributions, which constitute of a large fraction of very short bursts and a tiny fraction (less than 1%) contributes to about half of the total load, have been shown to be common in Internet files today(see [7] and references therein). By giving priority to the largest bursts under such workloads, OCSA negatively affects scalability of the system at heavy tailed burst sizes.

Chakchai So-In et al also proposed the enhanced (eOCSA) [6], which is similar to OCSA except eOCSA considers only one best mapping-pair either the least width or height of the subframe. Thus eOCSA lowers down the complexity whereas by considering all possible mapping pairs in OCSA, the complexity of OCSA increases with the resource allocation size. The eOCSA however fails to give priority to unscheduled bursts in the future frames and still is based on largest-first-approach. In general, the previous work on OCSA and eOCSA lack investigation on the algorithms' scalability.

In this paper, we first investigate the scalability issues of OCSA by comparing its performance under exponentially and heavy tailed bursts distributions. In particular, we first show that at high load, OCSA offers very poor performance in terms of delay under exponentially distributed workloads. To prevent starvation of small bursts under OCSA, we modify it and get iOCSA which gives priority to bursts that have been delayed up to a delay limit (delay threshold). Finally we propose a novel scheduling scheduling algorithm which we call **O**ne **C**olumn **S**tripping with **I**ncreasing **A**rea first mapping algorithm (OCSIA). As the name implies, OCSIA gives priority to smallest bursts first. Using numerical experiments we show that the proposed algorithm offers good performance even at overload conditions.

The rest of the paper is organized as follows: In the next section, the implementation of OCSA scheduling scheme. In Section 3, we discuss workload distribution and performance metrics, evaluate the performance of the scheduling algorithms and present detailed numerical results. In Section 4, we conclude the paper.

2 OCSA Scheduling Scheme

2.1 Overview of OCSA Algorithm

Under OCSA algorithm, data bursts at WiMAX base stations (BSs) are scheduled and mapped on the OFDMA WiMAX downlink subframe. Each subframe has 360 slots where each is capable of taking one unit of data. At each time unit (aka time slot) a frame can schedule and transmit at once up to 360 units of data. Upon arrival of bursts at the BS, the OCSA scheduler will map them according to the order of decreasing sizes until the frame is full. When the next largest burst doesnt fully fit in the remaining space in a frame, OCSA uses the technique of frame *optimization* which involves computing the unallocated space in the frame and selecting a smaller burst that fits the unallocated space in order to fully maximize system utilization.

Given the natural burst size in data networks, it is possible to have bursts sizes greater than 360 units. These bursts require more than a time slot to complete their service from the scheduler. When such a burst is encountered, OCSA scheduler serves it un-interrupted until it completed its service. Note that while servicing a large bursts greater than 360, similar new large bursts can be generated. Consequently, these are also scheduled first until they are over before any new short bursts receive service. In turn, depending on the workload distribution, the short bursts may experience considerable delays at the expense of favoring large data bursts.

2.2 Improved OCSA Algorithm

OCSA doesnt set any delay limit that a burst can experience. As a result, at very high loads and depending on workload distribution, short bursts may experience very high delays leading to unacceptable QoS performance. To avoid this excessive delays under OCSA, we propose an improved OCSA algorithm (iOCSA). That is, iOCSA sets a delay limit (D) for any burst in the system. This is achieved by giving the highest priority any bursts that have been delayed up to D time slots. They are therefore immediately scheduled before any other burst in the system. The exact value of D to use may be application dependent based on the underlying QoS requirements. That is, applications with stringent requirements require small D values. In summary, iOCSA is similar to the generic OCSA (i.e., it gives priority to large bursts). It additionally employs age based priority such that bursts that have been delayed for exactly delay limit D also have the highest priority regardless their sizes.

2.3 The Proposed Scheduling Algorithm

In this section, we propose a new OCSA based algorithm that we call **O**ne **C**olumn **S**tripping with **I**ncreasing **A**rea first mapping algorithm (OCSIA). OCSIA schedules bursts by giving priority to short bursts, this is analogous to size-based scheduling policy of Shortest Job-First (SJF). At any time slot, OCSIA scheduler maps bursts in the WiMAX frame according to the order of increasing sizes until the frame is full, i.e., starting with the smallest to largest.

The last smallest unmapped burst in OCSIA is likely to be larger that the remaining frame space. In such a case, the OCSIA scheduler splits the burst so that part of its burst is transmitted, and the rest is transmitted in the coming time slots based on OCSIA algorithm. We call the remaining unscheduled part of the burst a *partial burst*. In subsequent time slots, a partial burst is treated the same as any unscheduled burst. The partial burst may therefore be directly scheduled in the immediate slot or any of the future slots. It may also be further split in the subsequent time slots until it is fully scheduled or it may be scheduled in one time slot.

We implemented OCSA, iOCSA and OCSIA in Matlab in order to evaluate their performances and to compare their performances under varying workload distributions and load conditions. In the next section, we define workload distributions and performance metrics used in this paper, and present and discuss our findings.

3 The Performance Evaluation

3.1 Workload Distribution and Performance Metrics

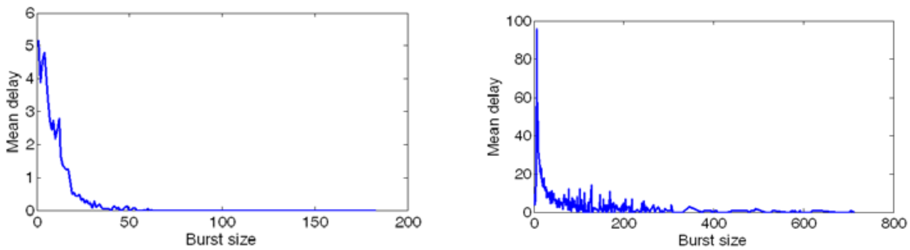
In this paper, we use exponential and bounded Pareto (BP) distributions to generate the workloads (bursts) of varying size behaviors. Exponential distribution is popularly used in practice to model samples and events for practical systems, whereas we use BP distribution to model the highly skewed workload of burst sizes which have recently been observed to model files sizes at various levels of network systems.

We shall represent BP distribution in short form as $BP(k, P, \alpha)$, where k and P are the minimum and the maximum burst sizes respectively and α is the exponent of the power law [7]. The Pareto distributions that emerge in computer system applications typically have $\alpha \in (0.9; 1.3)$. For performance evaluation of the algorithms, we generated exponentially distributed bursts sizes with parameter $\lambda = 0.04$, and heavy-tailed bursts sizes with BP distribution $BP(1, 720, 1.0)$. Both distributions are chosen to have equal mean value of 25. Note that the largest burst size for BP distribution is twice the frame size. Realistically, the largest burst size can be much larger that this but we dont have practical bursts' models based on realistic WiMAX networks. The two distributions are used in this paper to demonstrate the performances of the algorithms at typical distributions with high and low variability.

We draw our performance metrics from queuing theory. For instance, we define the term *delay* as the number of time slots a bursts spends in the queue before it is scheduled. We define *mean delay* as the mean of the number of time slots a burst of a given size remains in the queue before it is serviced. *Backlog* is the sum of the unscheduled bursts and *mean backlog* is the mean of unscheduled bursts. Lastly, as the term vividly implies, *percentage of scheduled bursts* is the percent of scheduled bursts at a given time slot. In the next section, we use these metrics to evaluate the performance of the scheduling algorithms presented in Section 2.3. To study the scalability of the scheduling algorithms we show the results at high load ($\rho = 0.9$) and overloaded ($\rho = 1.5$) conditions.

3.2 Performance of OCSA

Let's first look at the performance of OCSA under both, exponential and BP distributions. Our goal here is to show the weakness of OCSA in terms of penalizing short bursts for workload that have highly varying sizes. Due to limited space, we only show delay performances of OCSA at $\rho = 0.9$. We present more results of OCSA performance in terms of other metrics when we compare it with OSCIA in Section 3.4.



(a) The mean delay as a function of burst size under exponential distribution

(b) The mean delay as a function of burst size under BP distribution

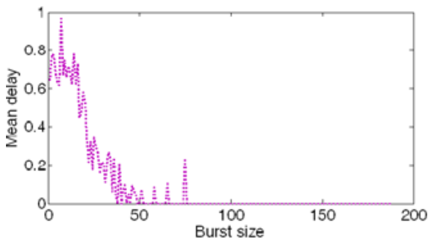
Fig. 1. The delay performances for OCSA at ρ=0.9 under exponential and $BP(1, 720, 1.2)$ distribution

Figures 1(a) and 1(b) show the mean delay as a function of burst size for OCSA under exponential and $BP(1, 720, 1.2)$ at load ρ =0.9 respectively. We can observe from the Figure that the shortest bursts are delayed on average by more than 90 time slots under BP distribution, which is significantly long and much longer compared to the case of exponentially distributed bursts size which exhibits a maximum average delay of only 5 time slots for short bursts. These results clearly show how OCSA penalizes short bursts for heavy tailed workloads. Since under heavy tailed distribution short bursts constitute a very large fraction of bursts this results also clearly demonstrate scalability issues of OCSA scheduler under these workloads.

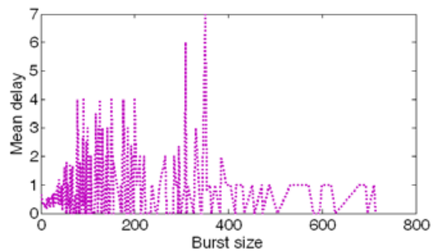
3.3 Performance of iOCSA

Recall that we propose iOCSA to limit the delay of bursts to avoid significant bursts delays under OCSA regardless of their sizes. In the results we show next, we use a delay limit $D = 5$ for iOCSA just for illustration purposes. Different applications require different practical D values to guarantee their acceptable quality requirements.

Figures 2(a) and 2(b) show mean delay as a function of burst size under exponential and $BP(1, 720, 1.2)$ distributions at $\rho = 0.9$ respectively. Comparing to the OCSA performance depicted in Figure 1 we observe much less mean delay under both workloads. Specifically, we can see from Figure 2(b) that the shortest bursts are now delayed by only 0.4 time slots on average, which is much shorter than 90 time slot as shown in Figure 1(b) for OCSA under BP distribution. Unexpectedly, Figure 2(b) shows that some bursts are delayed 7 time slots (higher than delay limit) on average. This is due to bursts greater than 360 which are generated and given service un-interrupted until all serviced. Compared to the results for OCSA in Figure 1, we can clearly see the performance benefits of iOCSA. It significantly reduces the mean delay of short bursts. In general



(a) The mean delay as a function of burst size under exponential distribution



(b) The mean delay as a function of time under BP distribution

Fig. 2. The delay performances for iOCSA at ρ=0.9 under exponential and $BP(1, 720, 1.2)$ distribution

however, regardless the distribution of the workload, iOCSA outperforms OCSA with negligible escalation of delay to other bursts.

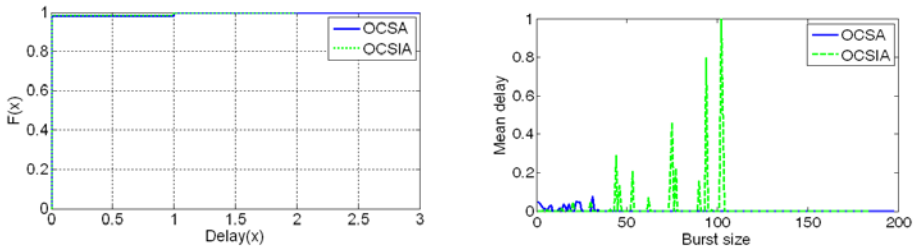
3.4 Performance of OCSIA

We now compare the performance of OCSIA to OCSA under exponentially and heavy tailed distributed workloads for different values of load ρ. As pointed earlier, for the purpose of studying scalability issues of OCSIA compared to OCSA, we also show the result for overloaded system (load ρ= 1.5) in addition to the results for $\rho = 0.5$ and $\rho = 0.9$. In this section, we present and discuss the performance of the algorithms in terms of all metrics defined in section 3.1.

Exponentially Distributed Workloads. A system is underloaded (Low load system), if utilization or total load of all resources in the system denoted as ρ, is less than 1. We first investigate the performance of OCSIA and OCSA at load ρ=0.5.

Figure 3 shows the delay performance for OCSIA and OCSA. Figure 3.3(a) shows the CDF of delay for OCSIA and OCSA. The results indicate that OCSIA offers delay of zero of all bursts. We note that more than 99% of the bursts experience no delay under OCSIA compared to about 98% of the short bursts with zero delay under OCSA. On the other hand, Figure 3.3(b) shows the mean delay as a function of burst size for OCSIA and OCSA. We can see from the figure that short bursts of less than 40 experience no mean delay and only large bursts above 40 experince low mean delay of less than 1 time slot under OCSIA. In contrast, short bursts under OCSA whereas no large bursts under OCSA experince any delay. This is due to the fact that OCSA favors large bursts and penalizes short ones whereas OCSIA favors short bursts and delays large ones slightly.

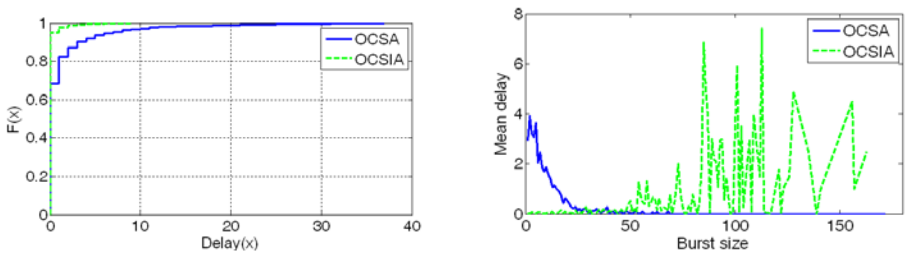
Figure 4 shows the delay performance for OCSA and OCSIA under exponentially distributed workloads at ρ=0.9. Figure 4(a) shows the CDF of delay for



(a) The cumulative distribution function of delay for OCSIA and OCSA

(b) The mean delay as a function of burst size for OCSIA and OCSA

Fig. 3. The delay performance for OCSIA and OCSA at ρ=0.5 under exponentially distributed workloads



(a) The cumulative distribution function of delay for OCSIA and OCSA

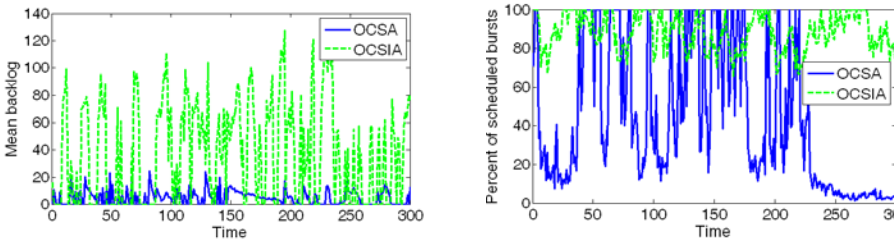
(b) The mean delay as a function of burst size for OCSIA and OCSA

Fig. 4. The delay performance for OCSIA and OCSA at ρ=0.9 under exponentially distributed workloads

OCSIA and OCSA which indicates that OCSIA offers lower delay for most of the bursts. For instance, one can quickly note that about 96% of the short bursts experience no delay under OCSIA compared to 68% of the short bursts with zero delay under OCSA. On the other hand, Figure 4(b) shows the mean delay as a function of burst size for OCSIA and OCSA.

It is evident from the Figure that short bursts experience no or lower mean delay under OCSIA than under OCSA. We also see that, as expected, large bursts experience some delay under OCSIA and no delay under OCSA. This is due to the fact that OCSA favors large bursts and penalizes short ones whereas OCSIA favors short bursts and delays large ones slightly.

Figure 5(a) shows the mean backlog as a function of time for OCSIA and OCSA. We observe that the mean backlog is highly bursty and attains large values for OCSIA and fluctuates around only 10 for OCSA. The mean backlog is computed from the unscheduled large bursts for OCSIA whereas it is from unscheduled short bursts for OCSA. On the other hand, Figure 5(b) shows the



(a) The mean backlog as a function of time for OCSIA and OCSA

(b) The percent of scheduled bursts as a function of time for OCSIA and OCSA

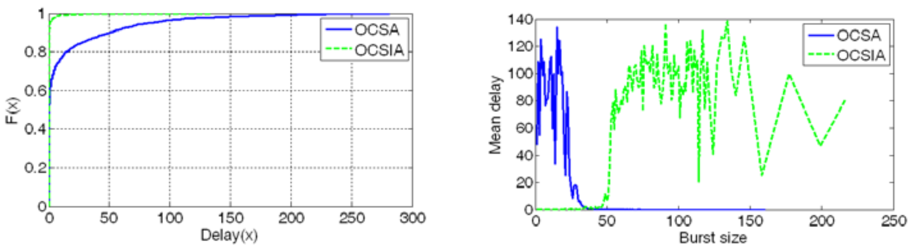
Fig. 5. The backlog performance for the OCSIA and OCSA at load ρ=0.9 under exponentially distributed workloads

percentage of scheduled bursts as a function of time for OCSIA and OCSA. As can be seen from the figure, the percentage of scheduled burst is much higher under OCSIA than under OCSA. OCSIA packs many short bursts in the frame. In contrary, when large bursts are serviced under OCSA, very few of these can be parked in the frame.

In real systems, it may happen that bursts arrive to the system at a higher rate than the rate at which they are serviced. This situation is referred to as overload condition where $\rho >1$. Practical systems are often over-provisioned to avoid poor performance at overload conditions, yet in rare cases such as those that lead *flash crowd* phenomena, legitimate requests severely overload systems. Scalable systems should be designed to handle overloaded situations. Therefore, it is useful to study scalability of OCSA and OCSIA at overloaded conditions. Next we compare the performance of OCSA and OCSIA algorithms at overloaded conditions.

Figure 6(a) shows the the CDF of delay for OCSIA and OCSA. We observe that the 95% of the short bursts were scheduled with zero latency for OCSIA compared to 64% of bursts (mostly large bursts) that were scheduled with zero latency for OCSA. This is due to the fact that OCSA gives priority to large bursts and penalizes short ones whereas giving priority to the short bursts, OCSIA delays large ones.

On the other hand, Figures 6(b) shows the mean delay as a function of burst size for OCSA and OCSIA. The figure has two two phases for each algorithm. For OCSIA, phase 1 represents short bursts from 0 to the 50 units which experience zero mean delay. In phase 2, from the 50th burst size to 240, we observe a sharp rise in delay with some large bursts are delayed at most 130 times on average under OCSIA. OCSA exhibits some what opposite pattern from OCSIA algorithm. In phase 1, we can observe from the figure that the shortest bursts are delayed by more than 120 time slots on average, which is significantly long and in phase 2 the large bursts had zero latency for OCSA. We explain this by the fact that OCSA gives priority to large bursts thus the short ones stay longer



(a) The cumulative distribution function of delay for OCSIA and OCSA

(b) The mean delay as a function of burst size for OCSIA and OCSA

Fig. 6. The delay performance for OCSIA and OCSA at ρ=1.5 under exponentially distributed workloads

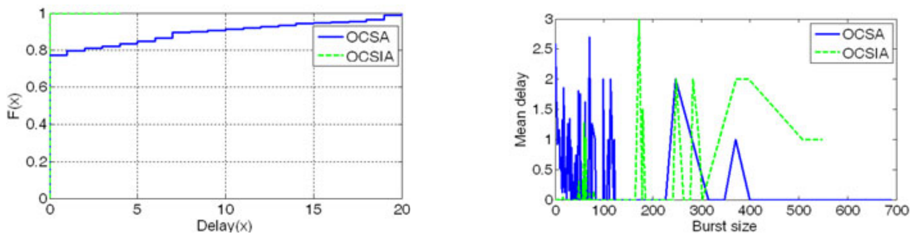
in the queue whereas OCSIA gives priority to the short bursts thus the large ones are delayed. Comparing to results for OCSIA and OCSA in Figure 4(b), Figure 6(b) also shows lower mean delays for short bursts for OCSIA.

The results in this section demonstrate that OCSIA indeed favors short bursts whereas OCSA favors large bursts. Even for exponentially distributed bursts, we observe from the figures that more bursts experience shorter latency under OCSIA than under OCSA. However, we also observe higher mean backlog under OCSIA than under OCSA but a larger fraction of scheduled bursts under OCSIA than under OCSA.

Heavy Tailed Workloads. In this section, we compare OCSIA with OCSA under heavy-tailed workloads. Despite lack of measurements to ascertain the exact model for bursts sizes in WiMAX networks, we strongly believe that realistic data burst sizes in WiMAX networks should exhibit similar behaviours as Internet traffic observed today at various levels of networked systems.

Figure 7 shows the delay performance for OCSIA and OCSA for $BP(1, 720, 1.3)$ at ρ=0.5. Figure 7(a) shows the CDF of delay under the BP distribution for OCSIA and OCSA. We observe that OCSIA slightly outperforms OCSA in terms of delay metrics. Firstly, we observe that more bursts experience no latency under OCSIA than under OCSA. Secondly, we also note in Figure 7(b) that mean delay experienced by bursts under OCSIA is shorter than under OCSA.

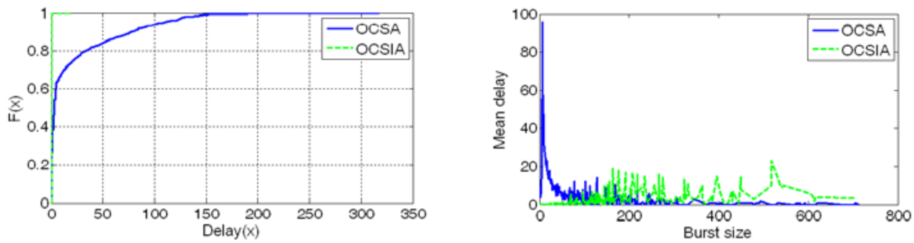
Figure 8 compares the delay performance of OCSIA with OCSA at load $\rho = 0.9$. We again observe that OCSIA clearly outperforms OCSA in terms of delay metrics. Observe that more bursts experience no latency under OCSIA than under OCSA. We also see that mean delay experienced by bursts under OCSIA is significantly shorter than under OCSA. Comparing this with the results under exponentially distributed workloads at similar load (Figure 4), we see a very high increase in mean delays for short bursts under OCSA, i.e., shortest bursts experience a mean latency of over 90 time slots under BP distributions compared to only 4 time slots under exponential distribution. This arises especially from



(a) The cumulative distribution function of delay for OCSIA and OCSA

(b) The mean delay as a function of burst size for OCSIA and OCSA

Fig. 7. The delay performance for OCSIA and OCSA at ρ=0.5 under $BP(1, 720, 1.3)$



(a) The cumulative distribution function of delay for OCSIA and OCSA

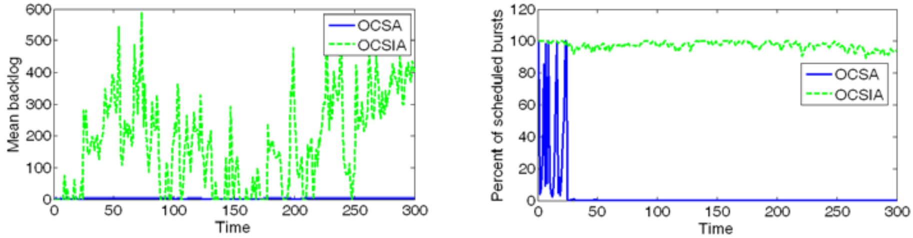
(b) The mean delay as a function of burst size for OCSIA and OCSA

Fig. 8. The delay performance for OCSIA and OCSA at ρ=0.9 under $BP(1, 720, 1.2)$

the fact that heavy tailed distributions like BP have more than 99% of their bursts short and a very tiny fraction of the largest bursts constitute more than 50% of the distributions' weight. Favoring large bursts given workload leads to severe starvation for short bursts under OCSA.

Figure 9b shows that consistently over 95% of bursts are scheduled at each time slot for OCSIA. This is compared to a very small value (negligible percentage) for OCSA. Obviously, the improvement in performance comes with an increase in backlog (due to the tiny fraction of largest bursts constituting more than 50% of the weight of the workload) as shown in Figure 9a. Since this backlog is contributed by a very tiny fraction, we can rightly conclude OCSIA significantly outperforms OCSA under heavy tailed workloads.

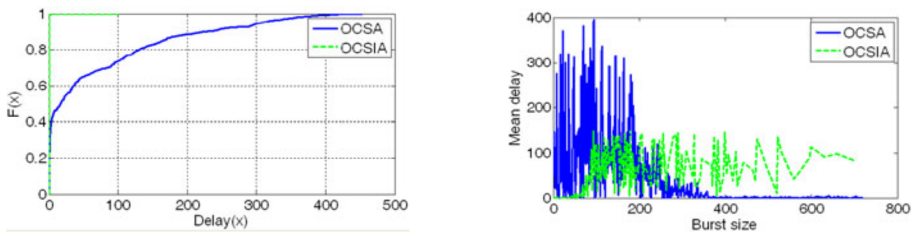
Figure 10 shows delay performances of OCSA and OCSIA at overloaded system. Both figures assert the results we observed for load $\rho = 0.9$. It is very intriguing to note that even at overloaded conditions, around 99% of the bursts are scheduled with zero latency under OCSIA compared to only around 35% for OCSA. We also note that short bursts (of sizes up to 50) experience zero or very



(a) The mean backlog as a function of time for OCSIA and OCSA

(b) The percent of scheduled bursts as a function of time for OCSIA and OCSA

Fig. 9. The backlog performance for OCSIA and OCSA at ρ=0.9 under $BP(1, 720, 1.2)$



(a) The cumulative distribution function of delay for OCSIA and OCSA

(b) The mean delay as a function of burst size for OCSIA and OCSA

Fig. 10. The delay performance for OCSIA and OCSA at ρ=1.5 under $BP(1, 720, 1.0)$

low latency under OCSIA compared to average latency of close or above 300 for short bursts of sizes up to 200 units for OCSA. The results for backlog and percentage of scheduled bursts for overloaded conditions show a similar trend as for load $\rho = 0.9$ but omitted here due to lack of space.

4 Conclusion

To summarize, we studied OCSA under exponential and heavy tail distributed workloads, proposed a variant of OCSA (iOCSA) that avoids starvation by giving priority to bursts that have been delayed for a given delayed limit, proposed a novel scheduling algorithm (OCSIA). We conducted numerical experiments to study the performance of the algorithms under exponentially and heavy tailed bursts size distributions. We performed our experiments at low, high and overloaded systems in order to investigate and compare scalability performance of the algorithms.

Our findings clearly show that iOCSA provides simple but efficient variant of OCSA that avoids starvation of bursts. The findings also show that the proposed OCSIA algorithm significantly outperforms OCSA algorithm in terms of scheduling more bursts with very low delay at any given time regardless of the workload distribution. We have observed that OCSA offers very poor performance and severely starves short bursts under heavy-tailed workloads, which is known to represent the actual models of realistic transfer sizes in networks and computer systems. It is also very interesting to observe that OCSIA performs very well even at overloaded conditions. It is intriguing that OCSIA successfully serves a significantly large fraction of bursts with zero latency (i.e., more than 99% and about 98% for heavy-tailed and exponentially bursts sizes). This clearly shows that OCSIA is a very scalable scheduling mechanisms for mobile WiMAX networks.

References

1. So-In, C., Jain, R., Al-Tamimi, A.: OCSA: An algorithm for Burst Mapping in IEEE 802.16e Mobile WiMAX Networks. In: Proc. the 15th Asia Pacific Conference on Communications (APCC 2009), Shanghai, China (October 2009)
2. Bacioccola, A., Cicconetti, C., Lenzini, L., Mingozzi, E.A.M.E., Erta, A.A.E.A.: A Downlink Data Region Allocation Algorithm for IEEE 802.16e OFDMA. In: Proc. Int. Conf. on Information, Commun. and Signal Processing (2007)
3. Ben-Shimol, Y., Kitroser, I., Dinitz, Y.: Two-Dimensional Mapping for Wireless OFDMA Systems. Proc. IEEE Transactions on Broadcasting 52, 388–396 (2006)
4. Desset, C., de Lima Filho, E.B., Lenoir, G.: WiMAX Downlink OFDMA Burst Placement for Optimized Receiver Duty-Cycling. In: Proc. Int. Conf. on Commun. 2007, pp. 5149–5154 (June 2007)
5. Ohseki, T., Morita, M., Inoue, T.: Burst Construction and Packet Mapping Scheme for OFDMA Downlinks in IEEE 802.16 Systems. In: Proc. IEEE Global Telecomunications Conf., Washington, DC, USA, pp. 4307–4311 (2007)
6. So-In, C., Jain, R., Al-Tamimi, A.: eOCSA: An algorithm for Burst Mapping with Strict QoS Requirements in IEEE 802.16e Mobile WiMAX Networks. In: Proc. IEEE Wireless Communication and Networking Conference (2009)
7. Rai, I.A.: QoS Support in Edge Routers, Ph.D. dissertation, Ecole nationale superieure des telecommunications, Paris, France (September 2004)