SELFLESS DISTRIBUTED CREDIT BASED SCHEDULING FOR IMPROVED QOS IN IEEE 802.16 WBA NETWORKS

C.Kalyana Chakravarthy[1] and Prof. P.V.G.D. Prasad Reddy[2]

[1]Department of CSE, M.V.G.R.College of Engineering,Vizianagaram,India
kch.chilukuri@gmail.com
[2]Prof. P.V.G.D.Prasad Reddy, Department of CS&SE,Andhra University College of Engineering,Visakhapatnam,India
prof.prasadreddy@gmail.com

ABSTRACT

Packet and flow scheduling algorithms for WiMAX has been a topic of interest for a long time since the very inception of WiMAX networks. WiMAX offers advantages particularly in terms of Quality of service it offers over a longer range at the MAC level. In our paper, we propose two credit based scheduling schemes one in which completed flows distributes the left over credits equally to all higher priority flows(FDCBSS) and another in which completed flows give away all the excess credits to the highest priority uncompleted flow(SDCBSS). Both the schemes are compatible with 802.16 MAC protocol and can efficiently serve real time bursty traffic with reduced latency and hence improved QOS for real time flows. We compare the two proposed schemes for their latency, bandwidth utilization and throughput for real time burst flows with the basic Deficit Round Robin scheduling scheme.

KEYWORDS

scheduling; quality of service; latency;

1. INTRODUCTION

IEEE 802.16 in PMP mode, defines five types of scheduling services[1] to support quality of service. They can be classified as Unsolicited Grant Services(UGS), Real-time Polling services(rtPS), Extended rtPS, non Real-time polling services(nrtPS) and Best Effort(BE).

Application of Unsolicited grant services (UGS) is Voice over IP (VoIP) without silence suppression. The mandatory service flow parameters that define this service are maximum sustained traffic rate, maximum latency, tolerated jitter, and request/transmission policy.

Applications of Real-time Polling service (rtPS) are Streaming audio and video, MPEG (Motion Picture Experts Group) encoded. The mandatory service flow parameters that define this service are minimum reserved traffic rate, maximum sustained traffic rate, maximum latency, and request/transmission policy.

Application of Extended real-time is VoIP with silence suppression. The mandatory service flow parameters are guaranteed data rate and delay.

Application of Non-real-time Polling service is File Transfer Protocol (FTP). The mandatory service flow parameters to define this service are minimum reserved traffic rate, maximum sustained traffic rate, traffic priority, and request/transmission policy.

Applications of Best-effort service (BE) are Web browsing, data transfer. The mandatory service flow parameters to define this service are maximum sustained traffic rate, traffic priority, and request/transmission policy.

Providing end-to-end QoS requires mechanisms in both the control plane and the data plane. Control Plane mechanisms allow the users and the network to negotiate and agree on the required QoS specifications and let the network appropriately allocate resources to each service. Data plane mechanisms are required to ensure the agreed-on QoS requirements by controlling the amount of network resources that each application/user can consume. Data Plane methods ensure the agreed-on QoS by classifying the incoming packets into several queues and allocating appropriate resources to each queue. Classification is done by inspecting the headers of incoming packets, resource allocation is done by using appropriate scheduling algorithms and buffer-management techniques for storing and forwarding packets in each queue. Two approaches called per-flow handling where we have a separate queue for each individual session or queue and aggregate handling where queues have packets from different flows exist. Both these methods offer their own advantages .While the later method reduces burden of state management and makes processing burden on nodes more scalable, the former method offers better QOS for the end users. Our method uses the first approach, since the main motive of our algorithm is to improve the Quality of service.

In WiMAX, the MAC layer at the base station is fully responsible for allocating bandwidth to all users, in both the uplink and the downlink. The only time the MS has some control over bandwidth allocation is when it has multiple sessions or connections with the BS. In that case, the BS allocates bandwidth to the MS in the aggregate, and it is up to the MS to apportion it among the multiple connections. All other scheduling on the downlink and uplink is done by the BS. For the downlink, the BS can allocate bandwidth to each MS, based on the needs of the incoming traffic, without involving the MS. For the uplink, allocations have to be based on requests from the MS.

Different connection management strategies have been proposed, but the most common one is of management connections first, real-time connections followed by non-real time connections and finally Best Effort connections.

In our paper, we propose and compare two credit based scheduling schemes Fair distributed credit based scheduling scheme and Selfless distributed credit based scheduling scheme. The first one is based on distribution of excess credits equally between all higher priority flows while the other proposed scheme is based on distribution of excess credits to the highest priority flow which is yet to be completed. The schemes are used to schedule flows between two classes of flows, real-time and non real-time flows.

We compare the two schemes in terms of the QOS parameters namely the throughput, bandwidth utilization, maximum latency etc., and observe that though the former one is based on fair scheduling, the latter in fact offers better performance under similar conditions compared to the basic DRR scheduling scheme.

2. PREVIOUS WORK

A significant amount of work has already gone into scheduling disciplines that provide delay guarantees and fairness.

Time stamp scheduler essentially uses the idea of assigning time stamps to packets and then transmitting the packets in some order that achieves fairness. WFQ [3] and WF^2Q [4] algorithms fall into this category. However, both of the schemes require a reference with the GPS server to be maintained. Variants of WFQ include Self-Clocked Fair Scheduling [5] and Virtual Clock [6], which do not need to maintain a reference GPS server and hence can compute the time stamp in a more efficient way. Though time stamp schedulers have good delay properties, their processing time is quite high.

Round-robin schedulers [7][8][9][10] are the other broad class of work-conserving schedulers. These schedulers typically assign time slots to flows in some sort of round-robin fashion.

Though they have better complexity compared to packet schedulers, however they have poor delay characteristics, particularly for packets of varying sizes. Several improvements have been proposed to improve the delay properties of the basic Round-robin scheduler. There is another class of algorithms that try to combine the tight delay bound of time stamp based schedulers and the low time complexity of round robin based schedulers. They usually adopt a basic round robin like scheduling policy plus time stamp based scheduling on a reduced number of units [11]. Bin Sort Fair Queueing [12] is based on arranging packets into different bins based on their time stamps and scheduling in a FIFO manner.

Stratified Round Robin [13] uses the round robin approach for inter-class scheduling and the time stamp approach for intra-class scheduling after grouping flows into respective classes.

Recently proposed algorithms like ADRR [14] enhance the deficit round robin scheduling discipline by taking into account the channel quality experienced by the transmitting node. The ADRR scheduler is designed to achieve performance isolation among links characterized by heterogeneous channel conditions.

In the DRR scheme, Stochaic fair queuing is used to assign flows to queues. For servicing the queues, Round-robin servicing is used, with a quantum of service attached to each queue. It differs from the traditional Round-robin in that if a queue is unable to send a packet in the previous round because a packet was too large, the remainder from the previous quantum is added to the quantum for the next round. Queues that are not completely serviced in a round are compensated in the next round. However, once a flow is serviced, irrespective of its weight, it must wait for $N-1$ other flows to be serviced until it is serviced again. Also, during each round, a flow transmits its entire quantum at once. As a result, DRR has poor delay and burstiness properties.

The Smoothed Round Robin discipline addresses the output burstiness problem of DRR. This is done by spreading the quantum allocated to a flow over an entire round using a Weight Spread Sequence. Although SRR also results in better delay bounds than DRR, the worst case delay experienced by a packet is still proportional to *N*, the number of flows.

3. FAIR DISTRIBUTED CREDIT BASED SCHEDULING SCHEME (FDCBSS)

In the FDCBSS, flows that complete first donate all their excess credits(debit), in addition to the quantum size equally between all the higher priority flows in the subsequent rounds until all the credits are exhausted.

Our model uses Inter-class scheduling for servicing the flows. It assumes fixed scheduling intervals between flows associated with a particular flow class. For each class F_k, the length of a scheduling interval is always 2^{k} slots. If a scheduling interval for F_k starts at slot t, the next scheduling interval for F_k starts at slot t + 2k, and so on. A flow is backlogged if it has not received it's fair share of bandwidth, i.e it still requires to be serviced in the next rounds. Backlogged flows are considered to be active. After every pending flow is serviced in the current time slot, clock time is t_c is incremented. Otherwise, t_c is advanced to the earliest time when some flow class becomes pending again. Also, in our model the bandwidth is assumed to be shared equally between the flows.

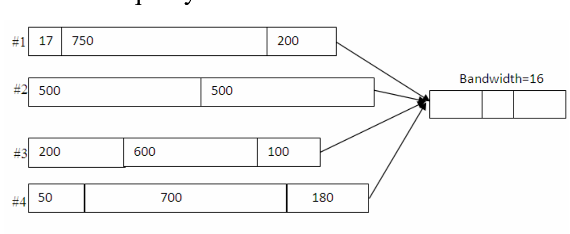

Figure 1.The Simulation Setup

The following scenario explains the operation of the FDCBSS. For simplicity, we have chosen the quantum size to be at least equal to the maximum packet size and the service pointer advances after each flow has been serviced.

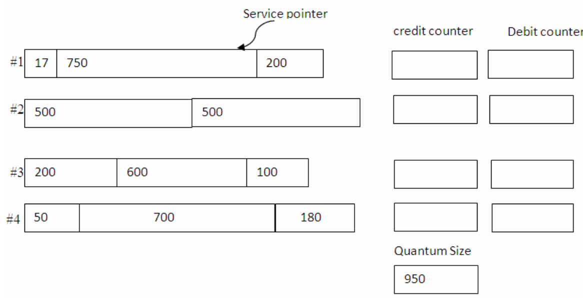

Figure 2.Beginning of Round one

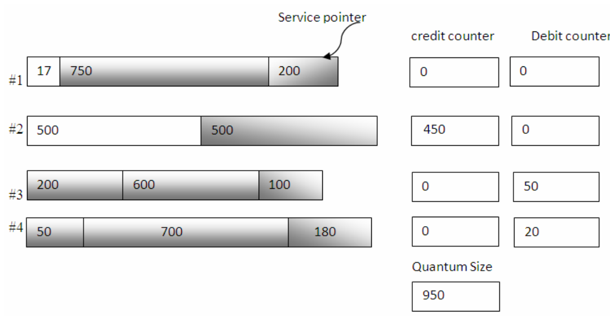

Figure 3.End of Round one

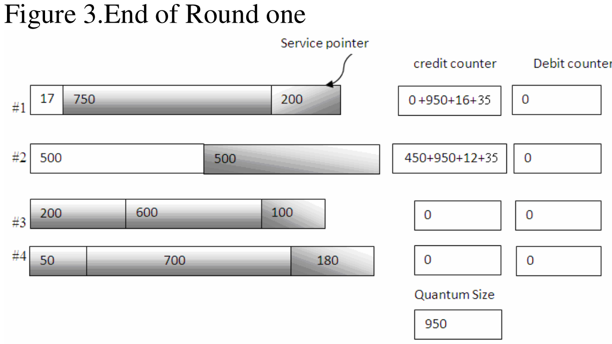

Figure 4.Beginning of Round two for FDCBSS

It can be noted that flows which are complete donate their debit to the highest priority flows yet to be completed, while other flows proceed the same way as in the DRR scheme[2].

```
For each flow do {
Initialize accumulated credit AC_i = 0;
Initialize accumulated debit DB_i = 0;
}
While true do
{
For each flow f_i do {
If(priority_i>p) and (DB_i>0) {distribute DB_i equally between flows        }
Compute available credit AV_i = (AC_i = AC_i + C_i);
For each packet in the flow PK_j do {

If available credit AV_i > packetsize in Bytes PS_j
Dequeue(PK_j);
AV_i = AV_i - PS_j;
Else
If packet size in Bytes PS_j > available credit AV_i
AV_i = AV_i + C_i;
}
If f_i complete and AV_i > 0, DB_i = AV_i
Priority_i =p;
}
}
```

Figure 5.The FDCBSS algorithm

4. SELFLESS DISTRIBUTED CREDIT BASED SCHEDULING SCHEME (SDCBSS)

In the SDCBSS, flows that complete first donate all their excess credits (debit), in addition to the quantum size in the next round to the highest priority flow that has not yet been complete. This enables the higher priority flows to only hold the credits as long as required and donate the remaining credits to the subsequent higher priority flow once it is complete.

The SCBSS differs from other scheduling schemes as in [15] where generally a completed flow distributes its credits continuously in subsequent rounds to the higher priority flows until it has no more credits to distribute.

The following scenario explains the operation of the SDCBSS. For simplicity, we have chosen the quantum size to be at least equal to the maximum packet size and the service pointer advances after each flow has been serviced.

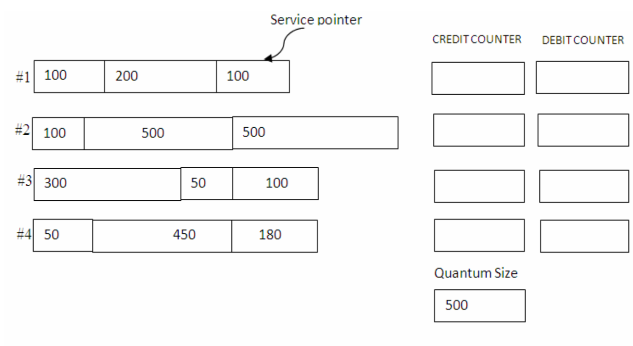

Figure 6.Beginning of Round one

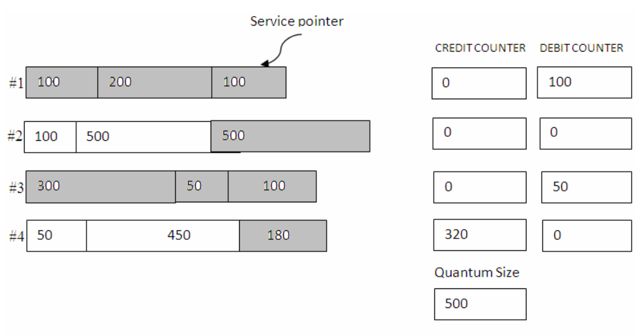

Figure 7.End of Round one

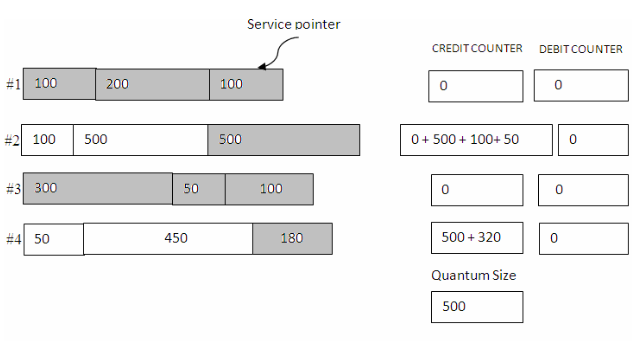

Figure 8.Beginning of Round two

```
For each flow fi do{
If(highest priority flow)
Compute available credit AVi = (ACi = ACi + Ci + debiti);
else
Compute available credit AVi = (ACi = ACi + Ci);
For each packet in the flow PKi do{
If available credit AVi > packet size in Bytes PSi
{
Dequeue(PKi);
AVi = AVi - PSi;
IF AVi > 0 debiti = AVi;
}
If packet size in Bytes PSi > available credit AVi
AVi = AVi + Ci ;
}
}
}
```

Figure 9.The SDCBSS algorithm

5. SIMULATION RESULTS

We use a custom simulator written in java. The simulation runs in two threads - the flow generator that generates packet and the scheduler that checks at every configurable scheduling period and schedules the packets. Both these modules can be run either concurrently or independently. Simulation has been carried out on 20 queues, each containing maximum packets of variable size, for different quantum sizes for 20 seconds and the results have been evaluated. The packets are generated according to Poisson arrival process. For our results we limited the number of flows so that the sum total of their minimum bandwidth requirements matches the maximum capacity of the network.

All flows are critical and are arranged in the decreasing order of their priorities. Our algorithm has shown reasonable improvement in terms of latency of critical flows, which makes it suitable for real time communications such as real time Video-on demand. If all latency critical flows meet the requirements, the maximum delay between latency critical flows should not exceed (n * s) + Max/B where n is number of latency critical flows, B bandwidth of the output line, s is maximum size of packet in a flow, Max is maximum quantum size.

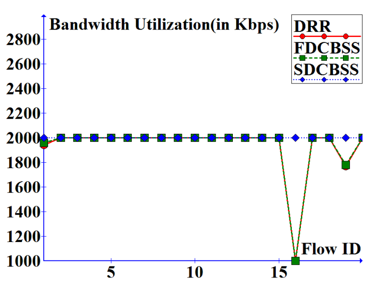

Figure 10.Flow ID VS bandwidth utilization

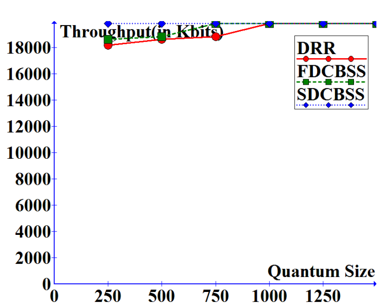

Figure 11.Quantum Size VS avg. throughput for 20 flows

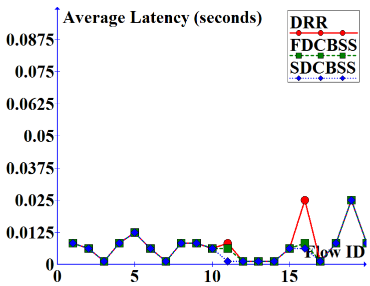

Figure 12.Flow ID VS latency

6. CONCLUSIONS

In our work, we have proposed two scheduling schemes FDCBSS and SDCBSS for scheduling real time flows. It was observed from the results that while both the schemes perform better compared with the Deficit Round Robin scheduling scheme, the latter is more suitable for real time flows under unsteady traffic conditions. In our method, any excessive idle bandwidth is reallocated to avoid wasting of available transmission capacity. However, in both cases, we assume scheduling under error free conditions.

Scheduling based on channel conditions is an active topic of research and the above scheduling schemes can easily be extended for example to consider back-off time required for retransmissions in erroneous channels at the link layer. With the awareness of channel condition and with knowledge of applications, schedulers can maximize the system throughput or support more users. Scheduling on Multiple Input Multiple Output channels with multiple antennas, scheduling on multi-hop networks for end to end service guarantees are also areas that need further improvement.

REFERENCES

[1] A. Sayenko, O. Alanen, J. Karhula and T. Hämäläinen," Ensuring the QoS Requirements in 802.16 Scheduling", Proceedings of IEEE/ACM MSWiM 2006, Torremolinos, Spain, Oct. 2006.

[2] M. Shreedhar and G.Varghese,"Efficient fair queuing using deficit Round Robin", in Proc. SIDCOMM '95 Boston, MA, Aug 1995.

[3] Demers,A.S Keshavand S.Shenkar,1989,"Analysis and simulation of a fair queuing problem", Proceedings of the Symposium and Communications Architectures and protocols, September 25-27,A.C.M NewYork, USA pp:1-12.

[4] Bennet , J.C.R and H.Zhang,1996,"WF2Q: Worst –case fair weighted fair queuing", Proceedings of the INFOCOM, March 24-28,San Fransisco,CA, pp:1-9

[5] Golestani,S.J.1994,"A self clocked fair queuing scheme for broad band applications", Proceedings of the 13th IEEE INFOCOM'94, Networking for Global Communications, June 12-16, Toronto, Ont, Canada, pp: 636-646.

[6] L. Zhang, "A new architecture for Packet switched network protocols", PhD dissertation, Massachesets Institute of technology, July 1989

[7] Lenzini, L., Mingozzi, E., and Stea, G. Aliquem: "a novel DRR implementation to achieve better latency and fairness at O(1) complexity," In IWQoS'02 (2002).

[8] "The Smoothed Round-Robin Scheduler", Paul Southerington, Member, IEEE, ECE742, 28 APRIL 2005.

[9] B. Bensaou, K. Chan, and D. Tsang, "Credit-based fair queuing (CBFQ): A simple and feasible scheduling algorithm for packet networks", IEEE ATM97 Workshop, pp. 589594, May 1997.

[10] Dessislava Nikolova and Chris Blondia, "Last-Backlogged First-Served Deficit Round Robin (LBFS-DRR) Packet Scheduling Algorithm", 15th IEEE International conference on networks,Nov.2007.

[11] Deng Pan, Yuanyuan Yang, "Credit Based Fair Scheduling for Packet Switched Networks", IEEE INFOCOM'05.

[12] S. Cheung and C. Pencea, "BSFQ: bin sort fair queuing," IEEE INFOCOM '02, pp. 1640-1649, New York, Jun. 2002.

[13] S. Ramabhadran, J. Pasquale, "Stratified round robin: a low complexity packet scheduler with bandwidth fairness and bounded delay," ACM SIGCOMM '03, pp. 239-250, Karlsruhe, Germany, Aug. 2003.

[14] Riggio, R. Miorandi, D. Chlamtac, "Airtime Deficit Round Robin (ADRR) packet scheduling algorithm," 5th IEEE International conference on Mobile Ad Hoc and Sensor Systems-MASS'08.

[15] Tsung-Yu Tsai, Zsehong Tsai, "Design of a Packet Scheduling Scheme for Downlink Channel in IEEE 802.16 BWA Systems", in WCNC'08 (2008).

Authors

Prof. P.V.G.D Prasad Reddy has a teaching experience of over twenty years. He is currently serving as the Registrar at the Andhra University. He has 14 publications in International Journals and 18 papers in conferences. His Research areas include Soft Computing, Software Architectures, Knowledge Discovery from Databases , Image Processing , Number theory & Cryptosystems.

C.Kalyana Chakravarthy has a teaching experience of over nine years and is currently working as Associate prof. in M.V.G.R.College of Engineering, Vizianagaram. He has been actively working on diverse areas of network caching, MANETs routing protocols, resource allocation and scheduling in WiMAX, Mesh networks