An Adaptive Bandwidth Request Mechanism for QoS Enhancement in WiMax Real Time Communication

Reetesh Mukul[1], Pradyumna Singh[1], D. Jayaram[2], Debabrata Das[1], Nayaka Sreenivasulu[2], Karnati Vinay[2], and Anoop Ramamoorthy[2]

[1] International Institute of Information Technology - Bangalore, 26/C Electronic City, Bangalore- 560 100, India.
e-mail: reetesh.mukul@iiitb.ac.in, pradyumna.singh@iiitb.ac.in, ddas@iiitb.ac.in
[2] Larsen Tubro Infotech Ltd., No. 2, Church Street, Bangalore- 560 001, India. e-mail: jayaram.d@lntinfotech.com

Abstract - The IEEE 802.16 standard for broadband wireless metropolitan area network supports real time and non-real time services. It has a provision to design new packet-scheduling algorithms according to requirements to support quality of service (QoS) for real-time services. Till now published literature on WiMax states that, a service station (SS) requests for bandwidth to a base station (BS) for already arrived packets at SS from users. The BS then allocates the bandwidth to SS according to priority-based request. In this paper we propose a novel adaptive-bandwidth scheduling algorithm at SS, for real-time polling services (rtPS), wherein the SS predicts the arrival of rtPS packets prior to the arrival and requests the BS for bandwidth in advance. It has been observed by analytical model and simulation that, this adaptive algorithm provided better results with respect to the number of packets waiting at SS and average delay as compared to the widely accepted weighted scheduling algorithm.

Index Terms—**Adaptive algorithm, QoS, WiMax, Real time**

I. Introduction

The IEEE 802.16 (referred as WiMax) for broadband wireless metropolitan area network (WMAN) is becoming popular mainly due to its open standard and support to quality of service (QoS) for different categories of services. A single cell in WiMax consists of a base station (BS) and multiple subscriber stations (SSs). The BS schedules the traffic flow in the WiMax i.e., SSs do not communicate directly. The communication between BS and SS are bidirectional i.e., a downlink channel (from BS to SS) and an uplink channel (from SS to BS). The downlink channel is in broadcast mode. The uplink channel is shared by various SS's through time division multiple access (TDMA). Figure 1 depicts the uplink and downlink subframes [1]. The subframe consists of a number of time slots. The duration of subframes, slots and the number are determined by the BS scheduler. The downlink subframe contains uplink map (UL map) and downlink map (DL map).The DL map contains information about the duration of subframes and which time slot belongs to a particular SS as the downlink channel. The UL map consists of information element (IE) which includes transmission opportunities.

WiMax supports all categories of services under real and non-real time communications. Real time caters to two types of traffic flow, i.e., unsolicited grant service (UGS) for T1/E1 type telecom voice and real-time-polling services (rtPS) for video streaming, teleconferencing as well as video conferencing. Similarly, the non-real time services is divided into two categories of traffic, i.e., non-real-time-polling services (nrtPS) to cater to file transfer and best effort (BE) services for non-delay sensitive communication, over Internet as e-mail, etc. The QoS is vendor specific and flexible for futuristic different services, wherein WiMax provides only signaling mechanisms and standard for UGS's QoS. It does not specify the admission control and packet scheduling algorithms at the BS as well as SS for rtPS, nrtPS and BE traffic. Out of these three undefined services, rtPS scheduling is more important, because it will cater to the real-time traffic, whose packet delay is very sensitive. It has been left to the traffic-designer to employ a particular admission control and packet scheduling algorithm for better performance of rtPS traffic.

There have been some studies on WiMax [2-6] QoS and its improvements depend on packet scheduling algorithms. In [3], authors have discussed uplink and downlink packet schedulers for bandwidth allocation process and admission control for QoS. Authors [4] have suggested changes in the MAC architecture to improve the throughput of services and have not discussed about delay. In [5], authors presented packet arrival and QoS issues for multimedia system. In [6], authors have done throughput evaluation for real-time communication but have not discussed the effect on delay due to the algorithm. In these packet scheduling algorithms SS sends a bandwidth request to BS after arrival of packet at SS. To the best of authors' knowledge, this paper proposes a novel adaptive scheduling algorithm for WiMax wherein a SS sends request for extra bandwidth beforehand to BS by speculating the rtPS traffic patterns. As the arrival patterns and service time are random, it is complex to predict the futuristic need of bandwidth at SS. Authors have presented by analysis and simulation that, the proposed algorithm is better than the existing mechanism on WiMax, in terms of delay and buffer requirement at SS.

1-4244-0340-5/06/$20.00 ©2006 IEEE

This paper is organized as follows. Section 2, presents the existing bandwidth request mechanism for rtPS and the proposed adaptive bandwidth request mechanism. In Section 3, mathematical analysis of the proposed solution is given. While, Section 4 contains the results of simulation and discussion, Section 5 contains the concluding remarks on contribution of this work as well as futuristic problems.

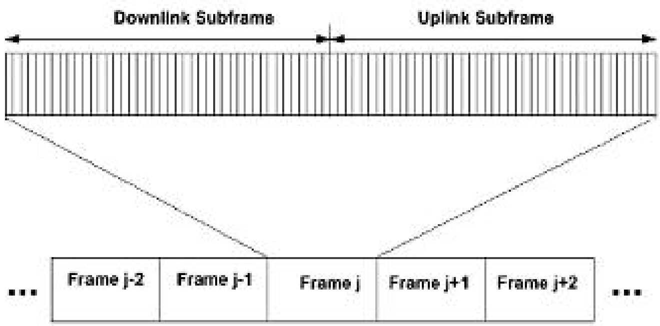

Fig. 1 802.16 Frame Format

II. Proposed rtPS Scheduler

In 802.16 the rtPS bandwidth request scheduler at SS demands time slots for data transmission. In response to the request by SS the BS will provide some time slots equal to or less than the requested time slot (i.e., according to availability of bandwidth). The time slots requested at a particular time depend on the total data in the rtPS queue at that time. Typically if the request for time slot is made at time, say t_0, and request is granted at time, say t_1, then $t_1 > t_0$. This means that there will be always delay incurred by the data packets because of request grant process. As the request was made at t_0, the BS will allocate bandwidth for rtPS traffic waiting till t_0. However, between t_1 & t_0 new rtPS data might have arrived since the time last request was made. For this new set of data, which have arrived between t_1 and t_0, the SS has to send fresh request for bandwidth allocation. This will incur extra delay. As rtPS services are very delay sensitive, we can alleviate these problems by scheduling an adaptive bandwidth allocation according to data flows in the rtPS queue. So we propose an rtPS bandwidth request model which has following features,

a. The rtPS queue at SS will demand time slots from BS, based on amount of data departures to be facilitated.

b. It will be adaptive in nature i.e., a SS will request time slot not only for present data in the queue but also on the data which will arrive in the queue in between the time the request was made and the corresponding time at which rtPS queue will be served . As the data arrival is random, this needs some sort of stochastic prediction of data that may arrive in the queue, in the aforementioned time interval (i.e., t_1 *and* t_0). The method adopted in this paper is "Differential time grant" method. Here we are estimating average rate of incoming data and duration of above time interval. These calculations will be used to estimate amount of the data that has arrived in this time interval.

The data packets coming to rtPS link is discontinuous in nature. The time slots in which it is served will be called 'ON' times while the time slots in which it is not served will be called 'OFF' times. Our algorithm does not presume any statistics of data entry in the system. So the computations are inherently deterministic in nature. This means that the results are applicable to any data arrival pattern. We have used Network Calculus [8] model to analyze our proposed rtPS scheduler (for modeling data flows).

III. Analysis of Proposed Scheduler

We have developed the analytical models in the following subsections for proposed adaptive algorithm to estimate, cumulative data flow at SS, data-flow equations, time-slot request function, time allocation for transmission time for a SS, evaluation of data arrival rate at SS, and estimation of adaptive time-slots. The analytical model that we have developed is generic in nature (applicable for all data arrival patterns). We compared the adaptive data flow with the flow based on length (referred as weight) of the rtPS queue.

A. Cumulative data flow

Let *A(t)* be the cumulative data arrivals into the rtPS scheduler. A model diagram of data arrival curve is shown in Fig. 2. In this graph we consider total data arrived in the rtPs queue till that time. The data arrival rate is assumed to have a mean of 1Mbps. Each time the maximum packet size is assumed to be 200 bytes.

B. Data flow equations

Let,

D (t) = Cumulative data departure from rtPS queue at SS

X (t) = Queue length in rtPS scheduler of SS

C = Maximum transmission rate at output link for rtPS

The data arrival curve as can be seen resembles a staircase function (Fig. 2). However here, the time width and height of each step is not constant. We assume that the departure rate is *C* or else it is zero. This means whenever data is served, it goes out at a rate *C*. For our system we have assumed *C* = 2Mbps, which is taken as the optimum rate at which a given SS will transmit. Thus data service curve is a piece wise linear curve with a flat segment between slant segments (Fig. 3). The time slots for which rtPS queue output link will be ON-OFF is stochastic in nature. For the steady state, the average rate of incoming data at the input port must be less than the average output rate of rtPS scheduler. Let the OFF time interval of rtPS output link be, (t_0, t_1), (t_2, t_3), (t_4, t_5), (t_6, t_7),... (t_{2n-2}, t_{2n-1}). Similarly the ON time of the rtPS will be, (t_1, t_2), (t_3, t_4), (t_5, t_6), (t_7, t_8),...(t_{2n-1}, t_{2n}).

We have assumed that the time origin of the system is at t_0. Further we are assuming the system is causal before time. That is $A(t_{0-}) = D(t_{0-}) = X(t_{0-}) = 0$. Let us denote the above shown ON time intervals as $P_1, P_2, P_3, \ldots, P_n$. Consider any such interval, for example $P_n = (t_{2n-1}, t_{2n})$. Let us define following two functions over P_n,

$$\min(P_n) = infimum\{t : t \in (t_{2n-1}, t_{2n})\} \quad (1)$$

$$\max(P_n) = supremum\{t : t \in (t_{2n-1}, t_{2n})\} \quad (2)$$

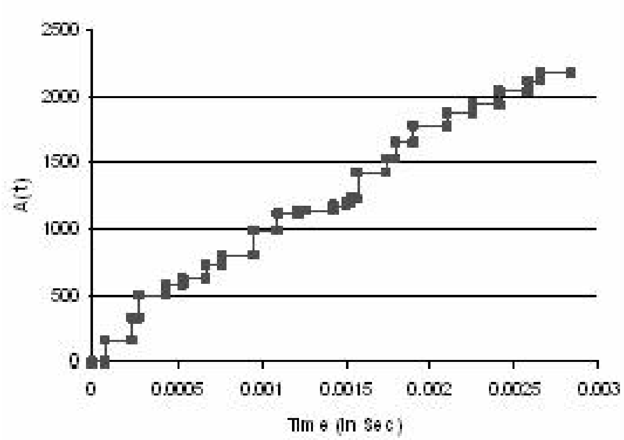

Fig. 2 : Data Arrival Curve $A(t)$ in Bytes in rtPS queue

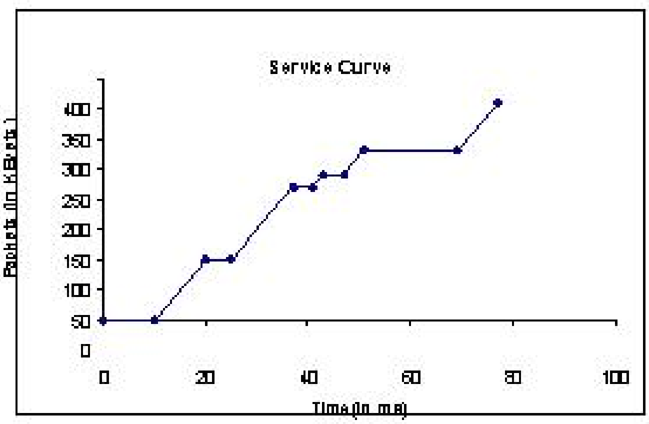

Fig. 3 : Service curve at rtPS scheduler

Data is flowing outside rtPS queue in 'ON' states only. We analyze the system for a given 'ON' interval. Let us define a function for $n > 1$, which considers net data flown out of rtPS queue at SS in last 'ON' time as,

$$F(t) = A(t) - D(max(P_{n-1})) \quad (3)$$

This equation simply means that during this 'ON' state we will consider data remaining in the system when the state started and new data arrivals since then. Using Reich's Equation [7], $X(t)$ and $D(t)$ over P_n can be given as,

$$X(t) = supremum\ (F(t) - F(s) - C(t-s)) \quad where\ s \in (t_{2n-1}, t_{2n}) \quad (4)$$

$$D(t) = infimum\ (F(s) + C(t-s)) \quad where\ s \in (t_{2n-1}, t_{2n}) \quad (5)$$

Similarly let us denote the OFF time intervals as $Q_1, Q_2, \ldots, Q_n$. Consider any such interval, for example $Q_n = (t_{2n-2}, t_{2n-1})$. Data over any such interval will only accumulate only. Thus, for $n > 1$,

$$X(t) = F(t) \quad (6)$$

$$D(t) = D(max(P_{n-1})) \quad (7)$$

C. Time slot request function

The time intervals P_n will have width ranging from 0 to say a maximum equal to T_{max}. This T_{max} will depend upon the BS's allocation mechanism of bandwidth to active SSs. This in turn will depend upon total number of end users in the system and the data flow emanating from them. Let us assume that the queue length of rtPS queue is L which is in some sense optimal. The optimality of it can be further evaluated based on the ON times we are getting in this system and the data flows entering the queue as well as type of data flow. Let us define a function $T_{slot}(x)$ to find out the time slot requested by rtPS scheduler,

$$
\begin{aligned}
T_{slot}(x) &= \text{time slot requested by the rtPS scheduler} \\
&= k * x, \qquad \text{when } x < T_{max}/k \\
&= T_{max}, \qquad \text{elsewhere.} \qquad (8)
\end{aligned}
$$

Where, the variable x will depend upon the load and other parameters of the rtPS scheduler. Also the function $T_{slot}(x)$ can have several variants (that is it can be defined in terms of x differently) provided it is constrained to lie below T_{max}. For our system we are taking the above definition of $T_{slot}(x)$, with $k = 1$. Consider the initial interval $Q_1(t_0, t_1)$. Let at some time α_1 in the interval Q_1, the rtPS queue demands for a time slot. Let the demand for time slot be given by (8), such that $x = X(\alpha_1)/C$, which is the queue length at time α_1 divided by link speed. The demand will be fetched till time t_2, where $t_2 = \max(P_1)$ (2). During the time duration $t_2 - \alpha_1$ data may arrive in the rtPS queue, which will be buffered. These data will have to effectively wait for another time slot (effectively because there can be the possibility that data arriving in this time may go earlier than the data already in the queue, however even then total amount of data waiting for next time slot will be increased by the amount of data coming in that interval). This leads to time lags which can be decreased by having greater time slots. Thus we have proposed a differential model in which next time when we request for time slot it will depend upon the hence coming data stream/packet also.

D. Estimation of time width

Let us consider that the times at which request are made be $\alpha_1, \alpha_2, \alpha_3, \ldots \alpha_x$. Now consider the time interval between α and the corresponding time till which it is fetched, i.e., if we call this variable β_i then $\beta_i = t_{i+1} - \alpha_i$. An estimate of β_i is to be made in terms of $\beta_{i-1}, \beta_{i-2}, \beta_{i-3} \ldots \beta_{i-x}$. Let us call this estimate $\tilde{\beta}_i$. Further let us define error in estimate as,

$$\delta_i = \tilde{\beta}_i - \beta_i \tag{9}$$

We are estimating β by taking two previous values,

$$\beta_i = \frac{(\beta_{i-1} + \beta_{i-2})}{2} - \delta_{i-1} \tag{10}$$

E. Estimation of data arrival rate

Similarly let us consider the quantity, v_i, which gives the data arrival rate

$$v_i = \frac{(A(t_{i+1}) - A(\alpha_i))}{\beta_i} \tag{11}$$

Similar to above process an estimate of v_i can be made in terms of $v_{i-1}, v_{i-2}, \ldots v_{i-x}$. Let us call this estimate $\tilde{v}_i$. Let us define error in this estimate as

$$e_i = \tilde{v}_i - v_i \tag{12}$$

Thus,

$$\tilde{v}_i = \frac{(v_{i-1} + v_{i-2})}{2} - e_i \tag{13}$$

F. Calculation of adaptive time slot

The calculation for time slot will be derived as,

$$x = \frac{(X(\alpha_i) + \tilde{v}_i * \tilde{\beta}_i)}{C}$$

(14)

$$\text{Time Slot} = T_{slot}(x)$$

(15)

IV. Results and Discussions

We have developed the simulation model according to adaptive analytical model for bandwidth request in the previous section. We simulated the network traffic to input link of rtPS queue at a given SS. Based on this simulation we evaluated cumulative data entry (*A(t)*, Fig. 2) into rtPS queue. Equations in Section 3 were used to model and find queue length of rtPS scheduler in SS and the net data flow till time *t* i.e., *D(t)*. The corresponding requests for time slots are also estimated. The input link of rtPS queue is assumed to receive data at an average rate of 1Mbps, unless otherwise mentioned. The output link of rtPS queue is served at a rate of C = 2Mbps. The average packet size is assumed to be 200 bytes. Simulation is done for 2000 packet arrival instants.

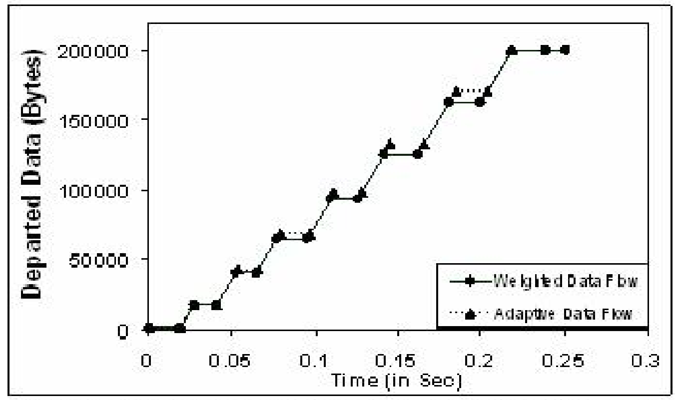

Fig. 4: departed data packets (in bytes) versus time (Sec) in a SS for adaptive and weighted

Figure 4 presents departed data packets (in bytes) versus time in a SS for adaptive and weighted (referred as non-adaptive algorithm). It is observed that our proposed adaptive algorithm (referred as adaptive flow) takes less time to transmit the entire arrived packet from a SS than weighted data for same packet arrivals. Thus it provides a better QoS for rtPS services with respect to delay. This is because, the departure in weighted flow considers only the amount of data in that rtPS queue at request time. However the departure in case of adaptive flow considers also the pattern of data that enters the rtPS queue at the time of request and additionally data between time of request to bandwidth allocation information arrival to SS from BS. Thus effectively, more data will depart from rtPS queue in case of adaptive flow in comparison to weighted flow as data will flow out in larger chunks in former case.

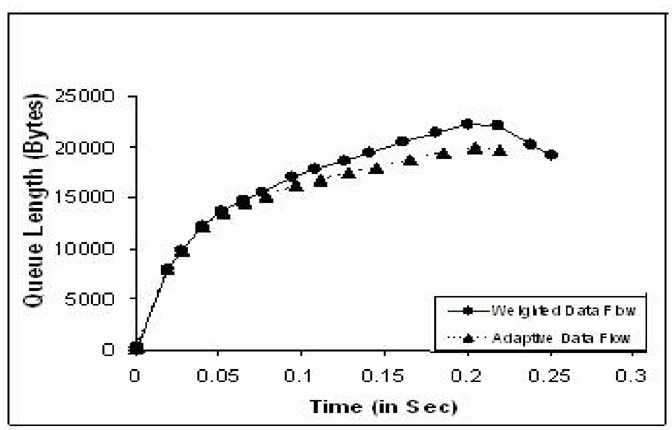

Fig. 5: average queue length (in bytes) versus time (Sec) for rtPS queue at SS, for weighted as well as adaptive flows

Figure 5, presents average queue length (in bytes) versus time (Sec) for rtPS queue at SS, for weighted as well as adaptive flows. This shows that queue length at SS is smaller in adaptive request scheduler. This is attributed to fast data flow out of rtPS queue due to adaptive flow mechanism as mentioned above. It can also be seen that data remains for longer time in case of weighted flow, for the same amount of data entering the rtPS queue. Moreover, adaptive algorithm needs less buffer size, as the queue length is less.

Figure 6, presents average delay (in micro-seconds) versus mean packet arrival rate in Mbps at SS, for adaptive and weighted flows. The mean data arrival rates range from 0.5 Mbps to 2 Mbps. At data arrival rate less than (say about ¾ times) output link speed (here output link speed 2Mbps) the adaptive curve has less delay than weighted curve. It has been observed that, at 1 Mbps input rtPS packets the average delay in adaptive flow presents decrease of 57% of delay in comparison to weighted flow. When mean arrival rate approaches output link speed the performance of weighted and adaptive request schemes become nearly equal because the output link has been saturated.

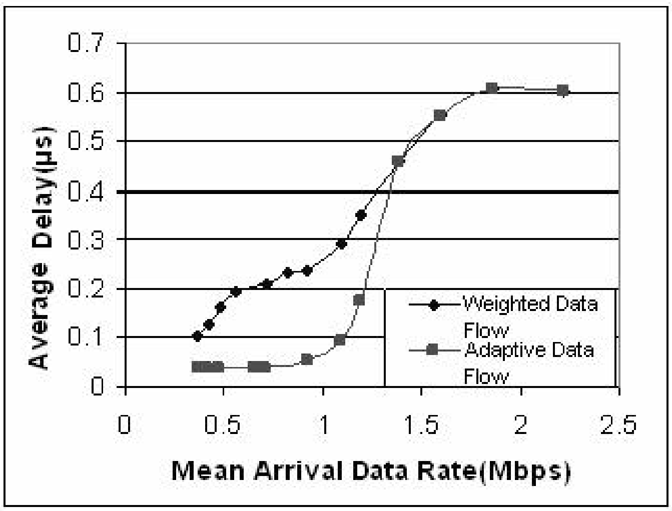

Fig. 6: average delay (in micro-seconds) versus mean packet arrival rate in Mbps at SS, for adaptive and weighted flows

Conclusion

The convergence of all kinds of traffic, will lead to a complex design scheduling algorithm for real-time communication. Our adaptive scheduling algorithm for WiMax's rtPS packets has shown remarkable decrement in delay. This is particularly suited for real time services. Moreover, this algorithm needs lesser size buffer at SS in comparison to conventional scheduling algorithms. Based on these calculations further optimization can be done for scheduling different sessions through rtPS queue.

References

[1] Draft of WiMax Standard, "IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air Interface for Fixed Broadband Wireless Access Systems", IEEE 802.16 forum.

[2] Carl Eklund, Roger B. Markas, and Kenneth L. Stanwood, "IEEE Standard 802.16: A technical overview of the wireless MAN Air In terface for Broadband Wireless Access", IEEE Communications Magazine, pp 98-107, June 2002.

[3] GuoSong Chu, Deng Wang and Shunliang Mei, "A QoS architecture for the MAC protocol of IEEE 802.16 BWA system" Communications, Circuits and Systems and West Sino Expositions, pp 435-439, IEEE 2002.

[4] Dong-Hoon Cho, Jung-Hoon Song, Min-Su Kim and Ki-Jun Han, "Performance analysis of the IEEE 802.16 wireless Metropolitan Area Network", Proceedings of the first International Conference on Distributed Frameworks for Multimedia Applications, IEEE Computer Society, 2005.

[5] Kitti Wongthavarawat and Aura Ganz, " IEEE 802.16 based last mile wireless military networks with quality of service support", pp 779-784, 2003

[6] Howon Lee, Taesoo Kwon and Dong-Ho Cho, "An efficient uplink scheduling algorithm for VoIP Services in IEEE 802.16 BWA Systems", IEEE, pp 3070-3074, 2004.

[7] Anurag Kumar, D. Manjunath and Joy Kury, *Communication Networking : An analytical approach*, Elsevier, 1st Edition, 2004.

[8] Jean-Yves Le Boudec and Patrick Thiran, *A theory of deterministic queuing systems for the Internet*, Open book, May 2004