This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2008 proceedings.

Design of a Packet Scheduling Scheme for Downlink Channel in IEEE 802.16 BWA Systems

Tsung-Yu Tsai
Networks and Multimedia Institute
Institute of Information Industry
Taiwan
tyt@nmi.iii.org.tw

Zsehong Tsai
Graduate Institute of Commun. Engineering
National Taiwan University
Taiwan
ztsai@cc.ee.ntu.edu.tw

Abstract- A packet scheduling scheme in the downlink channel for IEEE 802.16 is proposed. The scheme can be implemented in the base station and is compatible to the MAC protocol of IEEE 802.16. In order incorporate the properties of location-dependent and burst error of wireless channels, we design a novel packet scheduling algorithm named the Credit-Based Algorithm (CBA) to guarantee QoS of both real-time and non-real-time services. Simulation experiments are conducted to evaluate the performance of the proposed scheme. Based on the simulation results, we show the CBA scheme is capable of providing significantly improved system capacity, guaranteeing the packet delay for real-time services, and maintaining fair capacity allocation to non-real-time services.

I. INTRODUCTION

Currently, five various types of services, each supporting a different level of Quality of Service, have been defined in the latest release of the standard (IEEE 802.16-2005)[1] to satisfy a variety of applications supported in near future. However, the IEEE 802.16-2005 standard does not specify the packet-level scheduling mechanism for the base station. As a result, such down link packet scheduling scheme, one of the crucial technologies to provide the guarantee of QoS of the system level, may still be treated as an independent component of an 802.16 network system and is yet to be investigated from the system level.

In fact, although there have been so many research works [4-7] in the area of packet-level scheduling done in the recent years, a packet scheduling scheme specifically completely designed for efficient operations within an IEEE 802.16 MAC architecture is relatively few. Wongthavarawat and Ganz[3] completed an important reference work which provides a feasible solution to solve this problem. In [3], a classic scheduler-transmitter based two-tier hierarchical structure was shown to have the advantage of flexibility and was able to protect the QoS of the delay-sensitive traffic such as UGS connections or rtPS connections by simply assigning them higher priority. However, it also had the main drawback in the fairness between service types. As a result, it could sacrifice the QoS of the flows with lower priority when the flows with higher priority encountered bad channels. And the main cause of this draw back should be the lack of the cross-layer design: no feedback mechanism from the transmission operation at the physical channel to the MAC layer scheduling operations.

Based the performance of [3], one can also argue that if a classic packet scheduler such a WFQ[1] is employed in other 802.16 BS, it may also encounter a similar drawback. Unless, the cross-layer design concept is employed, we believed the overall system performance may always be limited with any independently operating scheduler. In this paper, we thus propose a new QoS-based packet scheduling scheme specifically designed for IEEE 802.16 networks. A novel packet scheduling algorithm is adopted in our scheme to consider the bursty and location-dependent error information regarding the wireless transmissions, and to satisfy the QoS requirement flows with different service types simultaneously.

The rest of this paper is organized as follows. In section II, we introduce our system model. Then we discuss our proposed packet scheduling scheme and the credit-based algorithm in deep in section III. In section IV, we evaluate the system performance by simulation experiment, the simulation results are shown in this section. Finally, we conclude this research in section V.

II. SYSTEM MODEL

A. Environment

In this study, we consider an IEEE 802.16-2005 network operating in OFDMA physical layer. The network architecture is PMP(point to multipoint), including one BS and several MSs which are in different locations. We address the scheduling of the downlink traffic. The duplex mode is TDD. The ratio of downlink and uplink subframes is assumed to be fixed and the sub-carrier permutation mode is kept not changed. Thus, the available slots for scheduling of downlink traffic can be regarded as constants.

B. Frame-by-frame Operation Scheme

We assume the IEEE 802.16 operates as a time-slotted system. For generality, we use “slots” as a general resource allocation unit and “frames” as a general time units for description of traffic characteristic and system capacity in the entire system. In this study, we assume our system has a

1525-3511/08/$25.00 ©2008 IEEE

total of B slots available for downlink traffic in a frame. Thus, the system capacity can be represented as B slots/frame.

Moreover, since the duplex mode is TDD, the transmission opportunity of a downlink packet is just the duration of DL subframe in each frame.

The proposed packet scheduler scheme called the CBA (Credit-Based Algorithm) scheme is also a time slotted scheme and schedules packets in a per-frame basis. That is, each frame forms a scheduling cycle, and each packet buffered in BS within a scheduling cycle are determined whether it should be scheduled to be transmitted in the next frame by BS. The result of scheduling in the scheduling cycle will be encoded in DL-MAP IE and broadcast to all the MS in the system at the beginning of the next frame.

Furthermore, we assume all the packet corruption due to channel error is always detected by the receiver and feedbacks NAK to the BS.

C. Traffic Profile

We divide all flows into two service groups: the real-time (RT) group and non-real-time (NRT) group, according to their fundamental differences of QoS requirement. This classification can easily made compatible with the five traffic types defined in IEEE 802.16-2005 standard.

Each flow associates to a flow profile for description of its traffic characteristic and required QoS. The flow profiles of RT flow and NRT flow are defined respectively as follows:

RT group: Real-time flows are typically delay-sensitive traffic such as VoIP and IPTV. A packet from RT flows is required to be transmitted successfully within its delay constraint or it is dropped by the system. However, certain loss rate may not degrade the application layer quality seriously and is tolerable for users. A triple $\{\delta_i, \lambda_i, D_i\}$ is used to describe the traffic characteristic of real-time flow i. Where δ_i is maximum burst size (normalized to slots), λ_i is the minimum reserved data rate (normalized to slots/frame), D_i is the maximum tolerable packet delay (in frame).

NRT group: Non-real-time flows are not sensitive to delay and jitter. The QoS matrix of non-real-time services is the average throughput. A parameter λ_j is used to describe the traffic characteristic of non-real-time session j, where λ_j is the minimum reserved data rate (normalized to slots/frame) of flow j.

D. Philosophy of Design

In a wireless environment, the information regarding the channel burst errors and location-dependent error is assumed available. A packet scheduler which is blind to the channel status of the receiver may send a packet repeat when the receiver incurs temporary burst error. It obviously harms the effective capacity of the system significantly especially when the total users increase. The philosophy of our design is thus to reduce the opportunity to transmit packet in error-prone channels, and then improve the efficiency of the channel usage.

To achieve this purpose, we introduce the concept of credits to represent protected capacity of a flow in the system. Those flows suffering temporary burst error may spend more credits than the protected ones. At this time, the overspent credits will be allocated to all the flows in the system. The more detail of the credit-based algorithm will be discussed in the next section.

III. THE CBA PACKET SCHEDULING SCHEME

A. System Architecture

Fig. 1 illustrate our proposed packet scheduling scheme operating in BS. It is composed of the following components. A classifier is at the frontend and is responsible for classifying packets arriving from the upper layer to the appropriate service group by some classification criteria. We assume that all the flows served have been admitted by an existing CAC(call admission control) mechanism and the task of our scheme is to schedule the mixture of packets among given sets of RT and NRT flows.

The remained components are the core of our scheduling scheme. They can be divided into two stages. The first stage is a virtual work-consuming system which plays as an underlying reference system. The second stage is a credit-based scheduler. It refers to the output of the virtual work-consuming system and makes the final decision that which packet should be scheduled to transmit in the next frame.

In the reminder of this section, we would address more about the operation and relationship of these components. Then, the proposed credit-based algorithm performed in the credit-based scheduler will be discussed in details. Finally, suggested settings of system parameters to achieve a fine conjugation with the overall design of our scheme will be available at the end of this section.

B. Virtual Work-consuming System

When a packet arrives from the upper layer, it first enters a virtual work-consuming system. The main purpose of the virtual system is to play as a reference system to our scheme. The actual conditions in the lower layer such as the channel status or the allocation of slots are transparent to this stage. It always assumes there is an error-free channel with fixed capacity $B^{'}$ in the lower layer. There is no constraint to the scheduling discipline adopted in the virtual system. In our packet scheduling scheme, we take WFQ [2] as the reference system.

After a new arriving packet scheduled by WFQ, the scheduling order is recorded in virtual queue for first time transmission (VQ-TX). There are

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2008 proceedings.

three virtual queue (VQ) in our system with strict priority. They are VQ for RT retransmission (VQ-RTRX), VQ for NRT retransmission (VQ-NRTRX), and VQ for first time transmission (VQ-TX) from the highest priority

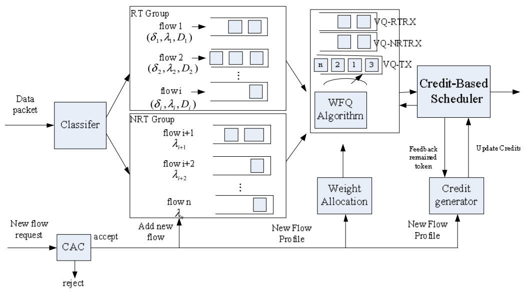

Fig. 1 The system architecture of proposed packet scheduling scheme

to the lowest priority. The three virtual queues are the inputs of the credit-based scheduler in the second stage. The Credit-based scheduler checks the packets in the virtual queues sequentially from the highest priority to lowest priority and determines whether to schedule it by credit-based algorithm. The packet not received successfully due to channel error is moved to one of the retransmission VQ according to its service group.

C. Credit-based scheduler

The Credit-based scheduler determines which packet should be scheduled to transmit in the next frame. The fundamental operation of the credit-based scheduler is as follows.

Assume that there are B slots available for downlink traffic each frame. Each flow i maintains a parameter *credits* c_i. The *credits* of every flow i has a fixed *credit incremental rate* r_i (slots/frame). At every beginning of a frame, the *credits* of flow i is increased by r_i slots. When a packet is transmitted, it should consume corresponding credits. Thus, r_i can be regarded as the protected capacity of flow i. Then, the basic principles of scheduling are as follows:

1. The packet which has sufficient credits has the higher priority, or it is assigned a lower priority.
2. For the packets with the same priority, the scheduling order is according to the order in WFQ scheduler

Based on the principles stated above, the processes of the packet scheduler in each scheduling cycle can be divided into two phases. The first phase is the *packet selection procedure*. The second is the *debt allocation procedure*. In the first phase, the scheduler checks the packets in each virtual queue. If a packet has sufficient credit, then it is scheduled for transmission; it will be skipped, otherwise. After all the packets are checked, if there are still available slots in the next frame, the scheduler enters the second phase. During the second phase, the scheduler checks the packets which are not scheduled in the first phase. If there are sufficient slots to transmit the checked packet, the packet will be scheduled, and the over-consumed credits are called *debt*. The *debt* is allocated to other RT flows in proportional to their weight. Here, we prefer to give RT flows more opportunity to increase their credits. Since if we clean RT packet as soon as possible, it is more likely to allocate remained resource to NRT flows, and meet the QoS requirement of both. The pseudo codes of the *packet selection procedure* and the *debt allocation procedure* are shown in Figure 2.

```
packet-selection-procedure(system_capacity){
  remained_slot←system_capacity;
  for(each flow j){
    block_j=0;
  }
  for(each virtual queue, from highest priority to lowest priority){
    while(packets not checked in the virtual queue){
        i← the flow the packet belong to;
        l←the size of the checked packet;
        if(remained_slot ≥ ⌈L_i + l⌉ − ⌈L_i⌉ and l ≤ c_i and block_i==0){
            schedule this packet in the next frame;
            remained_slot −= ⌈L_i + l⌉ − ⌈L_i⌉;
            c_i −= l;
        }
        else
            block_i=1;   /* the other packet of flow i is also skipped in this procedure */
    }
  }
  debt-allocation-procedure(remained_slot);
}
debt_allocation_procedure(remained_slot){
  for(each flow j)
    block_i←0;

  for(each virtual queue, from the highest priority to the lowest priority){

    while(packets not checked and not scheduled in the virtual queue){
        i←the flow the packet belong to;
        l←the size of the packet;
        if(remained_slot ≥ ⌈L_i + l⌉ − ⌈L_i⌉ and block_i!=1){
            schedule the packet in the next frame;
            debt←-1*max(c_i − l,−l);
            t_i −= l;
            remained_slot −= ⌈L_i + l⌉ − ⌈L_i⌉;
            for(all real-time flows k){
                c_k = max(c_k + w_k * debt / Σ_{j∈RT} w_j , C_k);
            }
        }
        else
            block_i←1;   /* the other packet of flow i is also skipped in this procedure */
    }
  }
}
```

Fig. 2 The pseudo code of the *packet selection procedure* and the *debt allocation procedure*

The provisioning of *credit incremental rate* alters the system performance significantly. We suggest that r_i is set to $\alpha_i * b_i$, where α_i is the protecting factor of flow i. It is a value larger than or equal to 1. b_i is the *minimum required capacity* of flow i. It is defined as the minimum required capacity need to be reserved for flow i to satisfy its QoS requirement. For RT flows, the QoS matrix is the tolerable delay, and for NRT flows, the QoS matrix is the minimum reserved data rate. b_i can be calculated as follows:

$$b_i = \begin{cases} \max(\dfrac{\delta_i}{D_i}, \lambda_i), i \in S_{RT} \\ \lambda_i, i \in S_{NRT} \end{cases} \quad (1)$$

where S_{RT} is the set of all RT flows, and S_{NRT} is that of all NRT flows.

Additionally, there should be an upper bound of the credits c_i, where we denote it by C_i.

IV.PERFORMANCE EVALUATIONS

A. Simulation environment

In this section, we evaluate the proposed CBA packet scheduling scheme by a sequence of simulation experiments. We start with an introduction to the testing environment.

The targeted testing system is based on a typical Mobile WiMAX network, consisting of a base station (BS) and several mobile stations (MS). The subcarrier allocation mode in the downlink is DL-PUSC, which is the default mode in typical Mobile WiMAX.

To appropriately characterize the nature of location-dependent and burst error of wireless channel, we model the channel by a two-state Markov chain. The channel states to different MS are assumed to be independent. A transition between the GOOD and BAD states can happen at the boundaries of each frame with the transition probability $P_{GB}(P_{BG})$ from the GOOD to BAD states(from BAD to GOOD states). If a packet is transmitted in a BAD frame, it will be received in error with probability 1; it will be received correctly, otherwise.

There are 3 typical network applications tested in this simulation experiments: IPTV, VoIP, and FTP. For IPTV and VoIP flows, we employ the traces of IPTV and VoIP traffic with a bit rate of 1510 and 84 kbps respectively (observed at the MAC layer). Moreover, we assume every mobile VoIP user has experiment a 100 ms constant latency between the other end and the BS. Furthermore, every user in the MS end has a jitter buffer. After the first packet received, the jitter buffer accumulates packet for 80 ms and then start the playout procedure. If a packet received after its expected playout time, it is regarded as meaningless and is dropped by the receiver.

The FTP traffic is generated by several simulated TCP sources. The version of the TCP is assumed to be *Reno*. The MSS is assumed to be 1500 bytes. Suppose that the payload of each packet is equal to MSS. We assume all MS are FTP clients and download files from the FTP servers in the Internet. The latency in the Internet from each FTP server to the BS is assumed to be fixed to 100 ms and no packet loss in the Internet.

There are two systems included in the simulation experiment for comparison purposes. Both adopt the same virtual work consuming system (WFQ). The major differences are at the second stages: one system simply adopts FIFO and the second adopts priority queue: with RT packets allocated higher priority and NRT packet allocated lower one. For convenience, we use the acronym "CBA", "FIFO", and "Priority" to stand for the proposed CBA scheme, the system with a FIFO queue, and the system with a priority queue in the performance figures.

B. Simulation Results

Experiment 1: The Performance in IPTV-dominant Environments

In the first experiment, we evaluate the performance of a system with only RT services using the CBA scheduling scheme.

We first test the performance by increasing the number of IPTV flows from 5 to 9. The profiles and associated algorithm parameters of each flow are shown in Table 1. This system can provide a peak downlink data rate of 15.552 Mbps. Additionally, the channel status of each flow is assumed to be independent and symmetric. The P_{GB} is 0.01 and the P_{BG} is 0.19. The ratio of the frame in BAD channel is 5%.

Fig. 3 shows the average packet loss ratio of all IPTV flows with different number of IPTV flows being served. Note that the packet loss ratio only accounts for those packets discarded due to delay violation. The buffer size is assumed sufficiently large so that no packet losses are due to buffer overflow.

One can find that when the traffic load becomes heavy (the number of IPTV flows is larger than 7). The average packet loss ratio of *FIFO* increases dramatically from %0.39 to %2.86. But the influence to CBA is limited and the average packet loss ratio keeps below 0.5%.

	MCS	Protected Capacity (Mbps)	Credit Rate r_i (credits/frame)	Max. Credits C_i	Tolerant Delay (frame)	Application & Service Group
Flow 1-9	64QAM 3/4	1.51	34.95	$5*r_i$	20 (100 ms)	IPTV (RT)

Table 1. The configuration of the flows in experiment 1

System Bandwidth	10 MHz
FFT size	1024
Subcarrier Allocation Mode in DL	DL-PUSC
Subchannels	30
Frame Duration	5 ms
Overhead Symbols per frame	DL:7 UL:4 TTG:1
Data Symbols per frame	36
DL/UL partition	DL:24 UL:12
Available slots in the DL	360

Table 2. The system parameter used in experiment 1

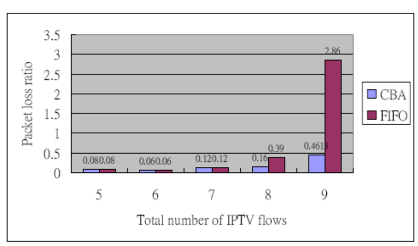

Fig. 3. The average packet loss ratio in experiment 1

Experiment 2: Test of Voice Quality

To further evaluate the real-time performance of CBA scheme, we test the quality of 35 VoIP in this experiment. The setting of parameters of flows and the tested system is shown in Table 3 and Table 4, respectively. Please note that the assumption of overhead in this experiment is larger than that of the previous experiment. In a network dominated by VoIP traffic, the number of users scheduled per-frame may be larger than that by other services, because of the small packet size and low data rate in VoIP flows. Thus, it may introduce more overhead, especially the overhead of MAP messages, which increases linearly as the number of scheduled users increases. This system can provide a peak downlink rate of 4.032 Mbps. Besides, the channel model of each flow is the same as that of the previous experiment.

For providing a more realistic evaluation of the quality of voice, we adopt Mean Opinion Score (MOS) [8] as the performance index which is a widely used benchmark for evaluation of voice quality. MOS is a value ranging from 1.0 to 5.0. In general, MOS larger than 4 means excellent quality, 3.0-4.0 means acceptable quality, below 3.0 means poor quality.

Table 5 shows the average MOS of all VoIP flows and the ratio of the time with MOS>3, denoted as *R(3)*. We can find that our proposed packet scheduling scheme outperforms the FIFO (with WFQ) design and the average MOS is 4.1 which is an excellent level of voice quality. The average MOS of the FIFO system is 2.4, which is much lower than that of our scheme and far from the acceptable level. About 93% of the time, the voice is in acceptable quality in our scheme, and only 22.75% of the time, the voice is in acceptable quality in the system with FIFO.

	MCS	Protected Capacity (Mbps)	Credit Rate r_i (credits/frame)	Max. Credit C_i	Tolerant Delay (frame)	Application & Service Group
Flow 1-35	16QAM 1/2	0.084	4.375	$8*r_i$	10 (50 ms)	VoIP (RT)

Table 3. The configuration of the flows in experiment 2

System Bandwidth	10 MHz
FFT size	1024
Subcarrier Allocation Mode in DL	DL-PUSC
Subchannels	30
Frame Duration	5 ms
Overhead Symbols per frame	DL:13 UL:6 TTG:1
Data Symbols per frame	28
DL/UL partition	DL:14 UL:14
Available slots in the DL	210

Table 4. The system parameter used in experiment 2

	CBA	FIFO
Average. MOS	4.1	2.4
R(3)	93	22.75

Table 5. The summary of voice quality in experiment 2

Experiment 3: Integrated Services with RT services and NRT services

In this experiment, we investigate the performance of a general integrated service network using the CBQ scheduling scheme. The simulation scenario is 4 FTP sessions as the background traffic. And we regulate the number of IPTV sessions from 5 to 9 to observe the effect to the system performance. The setting of parameters of flows is shown in Table 6. Other system parameters are the same as that in experiment 1.

We concern two properties of the NRT services. One is the *effective throughput* which is the average amount of data (slots) successfully received by the receiver of a flow in a frame. Another is the *Fairness Index* which is defined as follows:

$$\textit{Fairness Index} = \max_{i,j \in S_{NRT}} \left| \frac{S_i}{P_i} - \frac{S_j}{P_j} \right| \qquad (2)$$

where S_i and P_i is the normalized *effective throughput* and normalized protected capacity (slots/frame) of a non-real-time flow i respectively. The smaller the *Fairness Index*, the smaller the variation between the services received by each flow. In other words, smaller fairness index implies better fairness.

	MCS	Protected Rate(Mbps)	Credit Rate r_i (credits/frame)	Max. Credit C_i	Tolerant Delay (frame)	Application & Service Group
Flow 1-9	64QAM 3/4	1.51	34.95	$5*r_i$	20 (100 ms)	IPTV (RT)
Flow 10-14	64QAM 3/4	0.054	1.25	$100*r_i$	N.A.	FTP (NRT)

Table 6. The configuration of the flows in experiment 3

Fig. 4 and Fig. 5 show the average packet loss rate of all IPTV flows and the average throughput of all FTP flows with different number of IPTV flows. The average packet loss ratio suffered in the FIFO system is always higher than 25%. Obviously, the quality of the IPTV stream is far from the acceptable level. Though FIFO has achieved the highest FTP throughput in this experiment, it cannot guarantee the QoS of any RT flow. On the contrary, the *priority queue* uses all the system capacity to sustain real-time services. When the real-time traffic is heavy, the priority design uses most of the capacity to serve packets of real-time flows and it also starves the non-real-time flows. As a result, the average throughput is much lower than that of the other two, the usage of its channel capacity becomes inefficient, and the packet loss ratio of IPTV flows is also higher than that of CBA. Among all 3 systems, CBA is the system that achieves the most efficient usage of system capacity and successfully grants protection to non-real-time flows.

Since we are also interested in the efficiency of system channel capacity, we introduce a metric called *System Utilization*, for performance comparison. The *System Utilization* is defined as

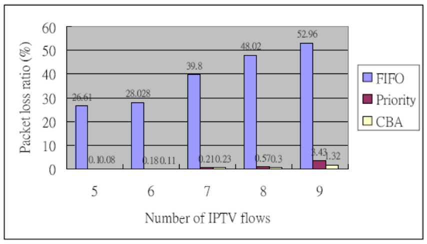

Fig. 4. The packet loss ratio of IPTV flows in experiment 3

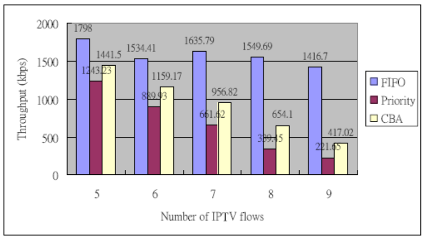

Fig. 5. The average throughput of FTP flows in experiment 3

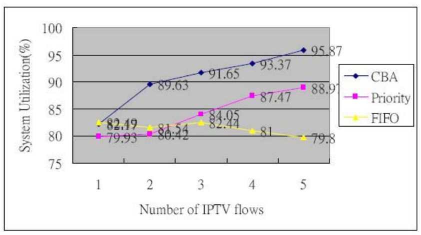

Fig. 6 The system utilization in experiment 3

	CBA	Priority	FIFO
Fairness Index	0.08	1.38	15.84

Table 7 The fairness index in experiment 3

follows:

$$System\,Utilization = \frac{the\ sum\ of\ normalized\ throughput(slots\,/\,frame)}{normalized\ system\ capacity(slots\,/\,frame)} \quad (3)$$

In Fig. 6, we can find that the *System Utilization* of the FIFO system in experiment 3 is limited to 81%. When the number of IPTV flows is larger than 8, the *System Utilization* of the priority design stops to increase and keeps at the value about 88%, and the quality of IPTV flows also decay rapidly at this point. On the other hand, the *System Utilization* of the CBA scheme is always much higher than the other two and achieves up to 95.87%, when the number of IPTV flows is 9. It is thus confident to say that the proposed CBA scheme does significantly improve the efficiency of the system channel capacity.

Table 7 shows the *Fairness Index* of these three systems when the number of IPTV flows is 9. Since the Fairness Index of CBA is only 0.08, which is much smaller that the other two, we conclude that CBA also significantly improves the fairness between non-real-time flows.

V CONCLUSIONS

In this paper, a two-staged packet scheduling scheme in downlink channel for IEEE 802.16 network is proposed. In order to handle the location-dependent and burst error in wireless link and to provide better QoS guarantee to both real-time and non-real-time flows, we have completed the design of a novel scheduling algorithm named Credit-Based Algorithm (CBA). Using the credit to represent the protected capacity of a flow, a debt allocation procedure is shown capable of reallocating the overspent credits(or capacities) consumed by a flow incurring temporary bursty error to other flows.

As illustrated in the pseudo code, the complexity of CBA should be relatively low. Comparing with other existing packet scheduling algorithms in wireless environment, it also has the advantages of low dependence to its reference system and does not need to couple with a channel prediction mechanism. These properties should have made the CBA packet scheduling scheme more practical to be implemented in a real base station.

According to simulation results, we show that the CBA scheme does improve the efficiency of capacity usage significantly and guarantees the delay for real-time flows, while effectively providing a fair allocation of the remained capacity to non-real-time flows. Thus, with the use of the CBA scheduler implemented in the base station, it should be much easier to meet the QoS requirement and other system performance objectives of both RT and NRT service groups, as promised in the original IEEE 802.16 standard.

ACKNOWLEDGEMENT

This work was partially supported by National Science Council, R.O.C., under Grants NSC-95-2221-E-002-198 and NSC-96-2221-E-002-157.

REFERENCES

[1] IEEE 802.16e-2005, “IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems – Amendment for Phisical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands,” Dec. 7, 2005.

[2] A. Parekh and G. Gallenger, “A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case,” *IEEE/ACM Trans. Networking*, June 1993.

[3] K. Wongthavarawat, A. Ganz, “Packet Scheduling for QoS Support in IEEE 802.16 Broadband Wireless Access Systems,” *International Journal of Communication Systems*, 2003.

[4] S. Lu and V. Bharghavan, “Fair Scheduling in Wireless Packet Networks,” *IEEE/ACM Trans. Networking*, 1999.

[5] T. S. Eugene Ng, I. Stoica, and H. Zhang, “Packet Fair Queueing Algorithms for Wireless Networks with Location-Dependent Errors,” *in Proc. INFOCOM’98*, 1998.

[6] F. Tsou, H. Chiou, and Z. Tsai, “WDFQ: An Efficient Traffic Scheduler with Fair Bandwidth Sharing for Wireless Multimedia Services,” *IEICE Trans. Communications*, 2000.

[7] Y. Liu, S. Gruhl, and E. W. Knightly, “WCFQ: An opportunistic Wireless Scheduler with Statistical Fairness Bounds,” *IEEE Trans. Wireless communications*, 2003.

[8] ”The e-model, a computational model for use in transmission planning,” *ITU-T Recommendation G.107*,2003