A Deficit Round Robin with Fragmentation Scheduler for IEEE 802.16e Mobile WiMAX[1,2]

Chakchai So-In, *Student Member, IEEE*, Raj Jain, *Fellow, IEEE*, Abdel-Karim Tamimi, *Member, IEEE*

Deficit Round Robin (DRR) is a fair packet-based scheduling discipline commonly used in wired networks where link capacities do not change with time. In wireless networks especially wireless broadband networks such as IEEE 802.16e Mobile WiMAX, the link capacity can change over time and also depends on the user location. Moreover, Mobile WiMAX allows packet fragmentation which violates the packet-based service concept of DRR. Therefore, the traditional DRR can not directly be used in such networks. Therefore, we introduce Deficit Round Robin with Fragmentation (DRRF) to allocate resources in a fair manner, while allowing for varying link capacity. Similar to DRR and General Processor Sharing (GPS), the DRRF achieves perfect fairness. DRRF results in a higher throughput than DRR while causing less overhead than GPS. In addition, we extend DRRF to support users with minimum reserved traffic rate, maximum sustained traffic rate and traffic priority.

Index Terms*—*Deficit Round Robin, Fragmentation, Mobile WiMAX, IEEE 802.16e, Scheduling, Resource Allocation, Fairness, QoS.

I. INTRODUCTION

To achieve high data rate, long distance coverage and mobility, IEEE 802.16e Mobile WiMAX standard [1] uses Orthogonal Frequency Division Multiple Access (OFDMA) technique. Basically, the entire channel is divided into multiple subcarriers. The number of subcarriers is proportional to the channel spectral width. These subcarriers are grouped into a number of subchannels. Then, each Mobile Station (MS) is assigned a group of subchannels for certain amount of time as shown by the two dimensional diagram in Fig. 1.

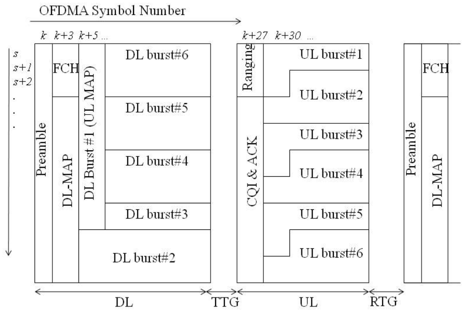

Fig. 1. A Sample OFDMA Frame Structure

In Fig. 1, the vertical axis is frequency or subcarrier or logical subchannel and the horizontal axis is time. The time is measured in units of OFDM (Orthogonal Frequency Division Multiplexing) symbol times. Mobile WiMAX uses a fixed frame-based allocation. Basically, each frame is of 5 ms duration [2]. It starts with a downlink preamble and Frame Control Header (FCH) followed by a downlink (DL) map and an uplink (UL) map. These maps contain the information elements that specify the burst profile for each burst. The burst profile consists of burst-start time, burst-end time, modulation type and Forward Error Control (FEC) used or to be used in the burst.

Fig. 1 also shows a ranging region in the uplink subframe. Ranging is used to determine the distance between the Base Station (BS) and MS so that the transmission start times at various stations can be properly synchronized. Ranging also helps to set the right transmit power level for each MS. CQI&ACK region is used to send Channel Quality Indication feedback and acknowledgements.

Bi-directional communication can be achieved by frequency division duplexing (FDD) in which uplink and downlink use different frequency bands or time division duplexing (TDD) in which the uplink traffic follows the downlink traffic in time domain. All scheduling schemes discussed in this paper can be used for both FDD and TDD systems. However, to keep the discussion focused, we use TDD throughput this paper.

Although the standard allows several configurations such as mesh networks and relay networks, our focus is only on point to multipoint network configuration. Thus, the resource allocation problem is basically that the BS is the single resource controller for both uplink and downlink directions for each MS. Each MS has an agreed quality of service (QoS) requirement that is negotiated between the BS and MS at the time of connection setup. The BS grants transmit opportunities to various MSs based on their bandwidth requests and QoS.

The resource allocation problem in IEEE 802.16e Mobile WiMAX is that of deciding the resource distribution among users. We basically focus on how to allocate the number of slots for each MS in each Mobile WiMAX frame (frame-based allocation). Each slot consists of one subchannel allocated for the duration of some number of OFDM symbols. The number of subcarriers in the subchannel and the number of OFDM symbols in the slot depend upon the link direction (uplink or downlink) and the permutation scheme used. For example, in Partially Used Sub-Channelization (PUSC) permutation scheme, which is commonly used in Mobile WiMAX, one slot consists of one subchannel over two OFDM

Manuscript received December 22, 2008, revised February 28, 2009. [1] This work was sponsored in part by a grant from Application Working Group of WiMAX Forum. [2] "WiMAX," "Mobile WiMAX," "Fixed WiMAX," "WiMAX Forum," "WiMAX Certified," "WiMAX Forum Certified," the WiMAX Forum logo and the WiMAX Forum Certified logo are trademarks of the WiMAX Forum. Parts of this paper were presented in the WiMAX Forum Application Working Group Meeting, Orlando, FL, Feb. 4-9, 2009.

C. So-In, R. Jain and A. Tamimi are with the Department of Computer Science and Engineering, Washington University in St. Louis, One brookings Drive, Campus Box 1045, Saint Louis, MO, 63130 USA (e-mail: cs5, jain and aa7@ cse.wustl.edu).

symbol periods for DL and one subchannel over three OFDM symbol periods for UL [1].

Mobile WiMAX supports several Modulation and Coding Schemes (MCSs), such as Binary Phase Shift Keying (BPSK) and several Quadrature Amplitude Modulation (QAM) schemes. BPSK results in 1 bit per symbol and is used for poor channel conditions. QAM schemes result in more bits per symbol and are used for reliable channel conditions. Since the MCS used for a mobile station depends upon the location of the mobile station and varies with time, the slot capacity (number of bits in the slot) is not constant. Given equal number of slots, mobile stations at different locations may be allowed to use different MCSs, resulting in different resource allocations.

In this paper we investigate two fair scheduling algorithms, namely, General Processor Sharing (GPS) [3] and Deficit Round Robin (DRR) [4] in the context of Mobile WiMAX networks. Each algorithm has its own pros and cons; for example, both GPS and DRR result in perfect fairness, i.e., equal throughput for contending users. However, since Mobile WiMAX also allows fragmentation, the GPS can utilize full frame resource allocation but create high overhead because the allocation is usually distributed equally for all MSs in each WiMAX frame. In contrast, because of the packet-based allocation framework, the DRR results in the least overhead but can leave some unused-space within the WiMAX frame.

Thus, to achieve full frame utilization as well as reduce overhead, we introduce Deficit Round Robin with Fragmentation (DRRF). In addition, we extend the DRRF to support other QoS parameters in Mobile WiMAX networks; namely, minimum reserved traffic rate, maximum sustained traffic rate and traffic priority [1].

This paper is organized as follows. In Section II, we describe a simple scheduling technique imitating General Processor Sharing – GPS along with Deficit Round Robin (DRR) and DRR with fragmentation consideration (DRRF). Section III presents a derivation of the number of slots for varying channel conditions and a simple algorithm for Max-Min fairness. In Section IV, we show how to derive the quantum size for users with minimum reserved traffic rate constraint and how to apply maximum sustained traffic rate and traffic priority parameters to DRRF. Section V shows simulation results. Finally conclusions and future work are discussed in Section VI.

II. FAIR SCHEDULING ALGORITHMS

In this section, we describe three scheduling algorithms. These are General Processor Sharing (GPS), Deficit Round Robin (DRR) and Deficit Round Robin with Fragmentation (DRRF) in the context of Mobile WiMAX networks.

A. General Processor Sharing (GPS)

The first algorithm, GPS [3], simply allocates the fair share to each mobile station queue regardless of the packet sizes. The GPS can achieve perfect fairness as well as full frame utilization. However, packet fragmentation can occur resulting in reduced goodput due to MAC header and fragmentation subheader overheads. For example, suppose if a single Mobile WiMAX frame has a capacity of 300 bytes and there are 4 active MSs with all packets of 125 bytes each, then only a 300/4 = 75-byte fragmented packet will be transmitted in every frame for each MS. Notice that, in practice a fair allocation for all MSs is infeasible with large number of MSs, say, more than the Mobile WiMAX frame capacity.

B. Deficit Round Robin (DRR)

The second algorithm, DRR [4], avoids packet fragmentation by scheduling only a full packet. If a packet will result in exceeding the fair share, the packet is not scheduled and the deficit (amount that would have been allocated) is remembered.

Table I shows an example of DRR. In this particular example, there are four MSs and each MS has only one flow and each flow is mapped to a single queue. The packet size is fixed to 125 bytes. Assuming the Mobile WiMAX frame capacity is 300 bytes. Then, to achieve a fair allocation, the fair share is set to 1/4th of 300 or 75 bytes. This is called the quantum size.

In this particular example, the packets are scheduled sequentially from each queue. We show four frames as an example. There are 2 rounds in the first and the third frames. There are no rounds in the second and the fourth frames. In Table I, notation *N/M* is used where *N* is the size of allowed transmission and *M* is the cumulative deficit counter value at the end of that round. Basically in each round, the deficit counter is increased by the quantum size, 75. The packets from queues with the deficit counters greater than the head packet size are scheduled. Again, in this particular example, only two packets in each frame are transmitted and result in 50 unused bytes. The packets being transmitted in each frame are shown in Table I. Notice that *N* is always 125 bytes.

C. Deficit Round Robin with Fragmentation (DRRF)

As we described above, one problem with DRR is that some space may be left unused in Mobile WiMAX frame since the next full packet will not fit in that space. The third algorithm, DRRF, avoids this problem. The DRRF is similar to the DRR but allows the fragmentation for the purpose of achieving full frame utilization, that is, in case there are some left-over spaces within a frame, the DRRF allocates those left-over spaces to some mobile stations.

We use the same example as that in DRR. Table II shows transmitted packet size and updated deficit counters. Again, we show four frames as an example (5 rounds). In each round, the deficit counter is increased by the quantum size, 75. Similar to the DRR, the packets with the deficit counter greater than the packet size are scheduled. However, to achieve full frame utilization, a fragmented packet is allowed as well. In other words, a fragmented packet with the deficit counter greater than the fragment size is also scheduled. In Table II, there are 2 rounds in the first two frames. In frame 3, there is no round and only one round for the fourth frame. In each frame, some fragmented packets may be also transmitted (we indicate these by bold numbers).

TABLE I
UPDATED DEFICIT COUNTER FOR DRR (TRANSMITTED PACKET SIZE /DEFICIT COUNTER)

FRAME	1^{st}			2^{nd}	3^{rd}			4^{th}
ROUND	1	2			3	4		
MS_1	0/75	0/150	125/25	0/25	0/100	0/175	125/50	0/50
MS_2	0/75	0/150	125/25	0/25	0/100	0/175	125/50	0/50
MS_3	0/75	0/150	0/150	125/25	0/100	0/175	0/175	125/50
MS_4	0/75	0/150	0/150	125/25	0/100	0/175	0/175	125/50

TABLE II
UPDATED DEFICIT COUNTER FOR DRRF (TRANSMITTED PACKET SIZE/DEFICIT COUNTER)

FRAME	1^{st}			2^{nd}				3^{rd}	4^{th}		
ROUND	1	2			3	4				5	
MS_1	0/75	0/150	125/25	0/25	0/100	0/175	**100**/75	**25**/50	0/50	0/125	125/0
MS_2	0/75	0/150	125/25	0/25	0/100	0/175	0/175	125/50	0/50	0/125	**75**/50
MS_3	0/75	0/150	**50**/100	**75**/25	0/100	0/175	0/175	125/50	0/50	0/125	0/125
MS_4	0/75	0/150	0/150	125/25	0/100	0/175	0/175	**25**/150	**100**/50	0/125	0/125

In the first frame, the first packet from MS_1 and MS_2, (each packet is 125 bytes) and the first fragmented packet from MS_3 (50 bytes) are transmitted (125 + 125 + **50** = 300 bytes). In the second frame, the last fragmented packet from MS_3, the first packet from MS_4 and the first fragmented packet from MS_1 are transmitted (**75** + 125+ **100** = 300 bytes). Two packets from MS_2 and MS_3 and the first fragmented packet from MS_1 and MS_4 are transmitted in the third frame and so on.

III. THROUGHPUT FAIR ALLOCATION

In wired networks, the scheduling algorithm (GPS and DRR) basically assumes that the link capacity is constant over time and location. However, it is not always the case in wireless networks and particularly Mobile WiMAX networks.

In addition, Mobile WiMAX allows various MCSs for reliable transmission and optimal throughput. The MCS levels are determined by the CINR (Carrier to Interference-plus-Noise Ratio) typically sent back over CQI&ACK channel and desired BLock Error Rate (BLER) [2].

As a result, to overcome the issue of varying link capacity, we apply the MCSs to calculate the proper fair share for GPS and the proper quantum size and update the deficit counters for DRR and DRRF. In other words, we basically use the number of *requested slots* not the number of bytes in the queue length (in downlink direction). The derivation of *requested slots* is shown below:

$$requested\_slots = \lceil queue\_length / MCS\_size \rceil$$

Here, *MCS_size* is the number of bytes per slot for the given MCS level. For example, for downlink PUSC mode with 10 MHz and 1024 FFT (Fast Fourier Transform), the number of subcarrier×symbol combinations per slot is 56. Of these 8 combinations are used as pilots leaving 48 combinations for data. With one bit per symbol coding, this results in 48 bits (or 6 bytes) per slot [1]. In case the queue length is 125 bytes, the number of requested slots is $\lceil 125/6 \rceil$or 21 slots.

For the uplink, requested slots can be derived from the bandwidth request. However, since there is no mechanism to send the information about individual packet sizes to the BS, DRR and DRRF can not be applied directly.

After the requested number of slots is determined for each MS, the per-frame fair share and quantum size are derived from the minimum of the total number of free slots per frame divided by the number of active MSs and the requested number of slots.

Notice that in Section II if the fair share and the quantum size are measured in slots, in those particular examples, each slot's capacity is one byte. However, different MCSs result in different number of bytes per slot and thus result in different fair share and quantum sizes in bytes. For the rest of the paper, we use the number of requested slots instead of requested bytes for all three schedulers in order to achieve *throughput fairness* for various channel conditions.

A. Max-Min Fairness Algorithm

In the discussion so far, we assume that all MSs have infinite traffic to send and can use the resources allocated to them. In other words, at scheduling stage all MSs always have packets waiting in their queues. However, in practice, the available traffic is finite. Some MSs may not have enough traffic and may not be able to use the fair share or some may have too much traffic. Our goal is to make a fair allocation among MSs. In the case where some MSs can not use the fair share, their left-over share should be fairly allocated to other MSs. This leads to the commonly used max-min criterion for fair allocation. An allocation is said to be max-min fair if it maximizes the allocation for the user that received the minimum. In Mobile WiMAX, the per-frame max-min fair allocation can be derived as follow;

$$Maximize\{Min(requested\_slots(i))\}$$

Here, again *requested_slots(i)* is derived from the queue length and MCS for downlink for the mobile station *i*. We use the MCS in order to derive the number of slots therefore finally each MS with different MCS will get the fair share of the throughput.

Fig. 2 shows steps in computing the Max-Min fair allocation. First step is to compute the number of requested slots from the number of bytes requested by an active mobile station and its MCS. Then, we sort the requested slots for all active MSs in ascending order (Step 2). In step 3, for each active MS the number of fair share slots is derived. Next, the number of granted slots, the minimum of number of fair share and the number of requested slots, is updated. Then, the number of free slots is updated. This loop continues until there are no more free slots or requested slots for all active MSs have been satisfied.

```
Calculate #requested_slots/frame for each active MS given its MCS //1st step
Sort active MSs in ascending order of active_MS_requested_slots //2nd step
FOR each active_$MS_i$ //3rd step
    Calculate #fairshare_slots for active_$MS_i$
    IF #requested_$slots_i$/frame < #fairshare_slots THEN
        #fairshare_$slots_i$/frame = #requested_slots
    END IF
    Update #granted_slots for active_$MS_i$
    Update #free_slots and exit if #free_slots == 0
END FOR
```

Fig. 2. Steps in simple Max-Min fairness algorithm

The results derived from Max-Min fairness algorithm, number of granted slots, are used to *as the actual quantum for DRR and DRRF*. For GPS, these granted slots are used as actual fair share allocations for each MS. Note that, *the quantum size and fair share may change over a frame period* according to the MCS level.

IV. DRRF EXTENSIONS

In this section, we describe how to apply the three extensions for Mobile WiMAX QoS parameters [1] to DRRF; namely, minimum reserved traffic rate, maximum sustained traffic rate and traffic priority.

A. Minimum Reserved Traffic Rate Extension

As described so far, the scheme ensures that all users will achieve fair throughput taking their channel conditions into account. However, by defining "*minimum reserved traffic rate*" or "r_{min}", some users may need to be favored over others. Note that users without minimum reserved traffic rates are treated as users with zero guaranteed rate.

We modified DRRF by updating the deficit counter. First similar to what we described in Section III, we derive the number of requested slots (*requested_slots*) from queue length and MCS. Then, we calculate number of slots required to guarantee the minimum reserved traffic rate (*rmin_slots*). The minimum of these two numbers (we call *pre_deficit.*) is used to update the deficit counter. Then, the number of free slots used to calculate the proper quantum is also updated accordingly. The equations below show how to derive these two numbers for mobile station *i*.

$$pre\_deficit(i) = \min[requested\_slots(i), r\min\_slots(i)]$$

$$r\min\_slots(i) = \lceil r\min(i) \times t\_frame / MCS\_size(i) \rceil$$

Here, *t_frame* is Mobile WiMAX frame size (5 ms). *MCS_size(i)* is the number of bytes per slot for the given MCS level of mobile station *i*.

Then, we use Max-Min fairness algorithm to calculate the proper quantum size to distribute the left-over slots fairly for both MSs without minimum reserved traffic rate and MSs with minimum reserved traffic rate but still need more slots than the total guarantee (see Section III.A).

B. Maximum Sustained Traffic Rate Extension

Maximum Sustained Traffic Rate, r_{max}, is used to control the peak rate. This rate excludes overheads such as MAC header [1]. We modified DRRF as follow:

First we derive the maximum number of slots required to meet r_{max}. We call this number *rmax_slots*. The derivation is shown below:

$$r\max\_slots(i) = \lfloor r\max(i) \times t\_frame / MCS\_size(i) \rfloor$$

Again, *t_frame* is the frame size. *MCS_size(i)* is the number of bytes per slot for the given MCS level of MS *i*. Note that instead of the ceiling function in case of *rmin_slots*, we use floor function.

Second, we apply Max-Min fairness algorithm without r_{max} constraint. Consequently *#fairshare_slots* is derived (see Fig. 2). Due to r_{max} constraint, some MSs may receive the fair share which is over the r_{max} limit. We limit those fair shares to *rmax_slots*.

Finally, in case there are some left-over slots due to r_{max} limitation, we apply another Max-Min fairness algorithm to distribute the left-over slots fairly among the flows which have no r_{max} constraint. Again *#granted_slots* derived from Max-Min fairness algorithm is used as the actual quantum.

C. Traffic Priority Extension

In Mobile WiMAX, traffic priority is one of the QoS parameters which differentiate the relative importance of flows with the same QoS class in which all other QoS parameters are identical. The standard does not specify how to differentiate between traffic priorities; however, it recommends that the scheduler should give lower delay and/or higher buffering preference for flows with high priority [1].

In a priority based system, flows of higher priority are serviced before those of the lower priority. Fairness implies that multiple flows of the same priority should get similar service. That is, if the system is overloaded, all flows of the lowest priority should be penalized equally. It is also a good practice to police the higher priority flows so that they do not overload the system and starve lower priority flows.

V. PERFORMANCE EVALUATION

In this section, we present simulation results of system throughput, percentage of overhead and fairness index for GPS, DRR and DRRF algorithms. We also show the results of DRRF with extensions. We consider only downlink resource allocation since we want to analyze the effect of fragmentation for various scheduling algorithms. For downlink, the queue size in number of bytes is translated to the number of requested slots. The analysis for uplink is very similar except that the bandwidth requests are used to derive the number of requested slots. However, the BS has no information about individual packet sizes. Therefore, both DRR and DRRF can not be used for uplink scheduling.

The simulation configuration and parameters follow the performance evaluation parameters specified in Mobile WiMAX System Evaluation document and WiMAX profiles [2]. These parameters are briefly summarized in Table III. With 10 MHz system bandwidth, 5 ms frame, 1/8 cyclic prefix and a DL:UL ratio of 2:1, the number of downlink symbol-columns per frame is 29 [2] (18 for uplink). Note that 1.6 symbol-columns are used for TTG (Transmit to Transmit Gap) and RTG (Receive to Transmit Gap). Of these, 1 symbol-column is used for preamble. In PUSC mode, there are 30 subchannels and each slot consists of one subchannel over 2 symbol duration. As a result, there are 30 × (28/2) = 420 downlink slots per frame.

Of these, Frame Control Header (FCH), DL MAP and UL MAP (repetition of 4 and QPSK1/2) and Downlink Channel Descriptor (DCD) and Uplink Channel Descriptor (UCD) overheads can approximately range from 51 slots to 195 slots (including 31 slots for DCD/UCD) in case of five and fifteen mobile stations, respectively [5]. Notice that the actual overheads depend on the number of actual burst allocations in both uplink and downlink and other management messages.

Table IV lists system throughput for each MCS. We show five and fifteen MSs (one flow per MS). In our analysis, interference is represented as a change of MCS. To keep it simple, the MCS level is constant over the simulation period.

TABLE III
PERFORMANCE EVALUATION PARAMETERS [2]

Parameters	Values
PHY	OFDMA
Duplexing Mode	TDD
Frame Length	5 ms
System Bandwidth	10 MHz
FFT size	1024
Cyclic prefix length	1/8
DL permutation zone	PUSC
RTG + TTG	1.6 symbol
DL:UL ratio	2:1 (29: 18 OFDM symbols)
DL Preamble	1 symbol-column
MAC PDU size	Variable length
ARQ and packing	Disable
Fragmentation	Enable
DL-UL MAPs	Variable

TABLE IV
DATA THROUGHPUT ANALYSIS (EXCLUDING DCD/UCD)

MCSs	Bit/ Symbol	Coding Rate	Bytes/ Slot	Throughput 5 MSs (kbps)	Throughput 15 MSs (kbps)
QPSK¾	2	¾	9	5,472	4,780
16QAM½	4	½	12	7,296	6,374
16QAM¾	4	¾	18	10,944	9,561
64QAM⅔	6	⅔	24	14,592	12,748
64QAM¾	6	¾	27	16,416	14,342

A. Simulation Configurations

We used a modified version of the WiMAX Forum's ns-2 simulator [5] in which a mobile WiMAX module has been added [6]. Fig. 3 shows the simulation topology. The link between BS and MSs is the bottleneck. Other simulation parameters are as shown earlier in Table III. At the BS, there is one queue for each MS and each queue is 100 packets long.

There are two main simulation configurations in order to show the fairness among MSs with different MCSs and the effect of overhead versus system utilization for GPS, DRR and DRRF. In addition, we simulated three more configurations to show DRRF with minimum reserved traffic rate, maximum sustained traffic rate and traffic priority extensions.

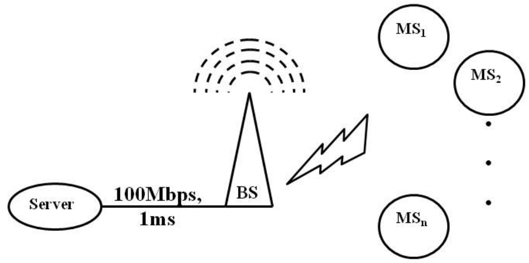

Fig. 3. Simulation Topology

In the first scenario, all users have zero minimum reserved traffic rates, no maximum sustained traffic rate limitation and no distinction of traffic priority. In order to show the effect of the scheduler, we used Constant Bit Rate (CBR) traffic flows with 3 Mbps source rate. There are five MSs. To show the effect of throughput fairness, each MS has only one flow and is mapped to one of the MCSs shown in Table IV.

In the second scenario, to show the effect of the overhead versus the system utilization, we simulated fifteen MSs. Again, each MS has only one flow and there are no constraints on any extensions. We used CBR traffic flows with 500 kbps. The modulation and coding scheme was fixed to QPSK3/4.

Note that five flows (15 Mbps) and fifteen flows (7.5 Mbps) can saturate the link capacity so that the fairness is exercised (there are dropped packets due to buffer overflow at the BS).

In addition, with minimum reserved traffic rate extension, the simulation configuration is similar to that in the first scenario, five MSs and 3 Mbps each. However, we fixed the MCS to QPSK3/4 and a minimum reserved traffic rate of 2 Mbps for MS_1.

With maximum sustained traffic rate extension, similar to the first scenario, five MSs with 3 Mbps each used an MCS of QPSK3/4. In addition, we set the maximum sustained traffic rate of MS_1 to 500 kbps.

Finally, with traffic priority extension, again similar to the first scenario, we simulated 5 MSs, 1.5 Mbps each with an MCS of QPSK3/4. However, we set the traffic priority of MS_1 and MS_2 at 2 and MS_3 to MS_5 at 1. The expected fair throughputs of the first two MSs are 1.5 Mbps and the fair share of the left-over capacity for MS_3 to MS_5.

In fairness simulations, we do not use TCP flows because the slow-start feedback mechanism of TCP has a significant impact on the resource usage and it is difficult to isolate it from the effect of scheduling algorithm. Note that though we use five and fifteen flows to show the effect of overhead and fairness, with more MSs or higher number of flows, intuitively the effect of overhead will be more obvious.

All simulations were run from 0 to 10 seconds with 5 seconds of traffic duration. Flows start at 5 seconds end at 10 seconds. There are ranging, registration and connection setup processes during the first 5 seconds. The packet size is set at 1500 bytes.

The percentage of overhead was calculated from MAC header and fragmentation subheader. We do not consider the optional packing subheader and other subheaders. In term of fairness, we used Jain Fairness Index [7], which is computed as follows:

$$f(x_1, x_2, \ldots, x_n) = (\sum_{i=1}^{n} x_i)^2 / n\sum_{i=1}^{n} x_i^2$$

Here x_i is the throughput for ith MS and there are n MSs or n flows. In our simulation, n is 5 and 15.

B. Simulation Results

In this section, we show system throughput, percentage of overhead and fairness index of GPS, DRR and DRRF. We also show the results of DRRF with extensions. The throughput is similar to that obtained by our numerical analysis (GPS and DRRF). Note that the actual throughput varies because of the variation of the actual number of burst and Downlink Channel Descriptor (DCD) and Uplink Channel Descriptor (UCD) and other management messages being transmitted as well.

For the first scenario, since the number of slots derived from the queue length and its MCS level is used, the results show that all three algorithms can achieve perfect *throughput fairness*: Jain Fairness Index is *1*. The average throughput of GPS and DRRF is around 1.78 Mbps. For DRR, the average throughput is around 1.47 Mbps due to the frame underutilization.

In the second scenario, the overhead versus system utilization, since DRR only sends a full packet, the percentage of the overhead is the lowest (0.37%). In other words, there are no fragmented packets. Table V shows that the overhead of GPS is the highest because GPS has no consideration of the packet size (allocated portion of the packet is transmitted to each MS in each frame). The overhead of DRRF is close to that of DRR because it favors full packet transmission with fragmentation allowed.

In terms of system throughput, since the fragmentation is not allowed for DRR, the system throughput of DRR is the lowest: 2.42 Mbps versus 4.65 Mbps for GPS and DRRF. To sum up, *DRRF gives higher system throughput than DRR and has less overhead than GPS.*

TABLE V
SYSTEM THROUGHPUT VS. PERCENTAGE OF OVERHEAD

Algorithms	**System Throughput (kbps)**	**%Overhead**	**Fairness Index**
GPS	4,658	4.62	1.00
DRR	2,418	0.37	1.00
DRRF	4,651	0.71	1.00

In addition, for DRRF with minimum reserved traffic rate extension, the result shows DRRF can support the minimum reserved traffic rate. Fig. 4 shows the throughput for all five MSs. In general, MS_1 receives the minimum bandwidth at 2 Mbps and the left-over bandwidth is distributed fairly among MS_1 and others. Notice that to meet the 2 Mbps rate guarantee, the BS also needs to take the overhead into account: 6 byte MAC header and 2 byte fragmentation subheader.

For DRRF with maximum sustained traffic rate extension, Fig. 5 shows that the average throughput of MS_1 is 492 kbps and the left-over bandwidth is distributed fairly among others (fairness indexes are all *1*, each flow throughput except MS_1 is around 1.2 Mbps).

Finally, for DRRF with traffic priority extension, Fig 6. shows both flows of priority 2 get equal throughput, 1.5 Mbps, while the three flows of priority 1 get a fair share of the left-over capacity, 0.7 Mbps.

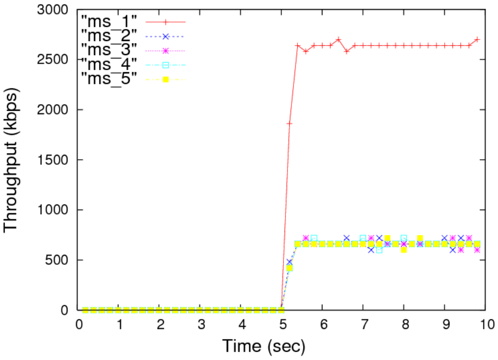

Fig. 4. Throughput of DRRF (MS_1 with 2Mbps r_{min})

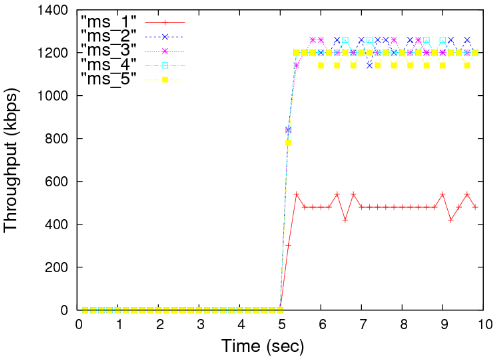

Fig. 5. Throughput of DRRF (MS_1 with 500 kbps r_{max})

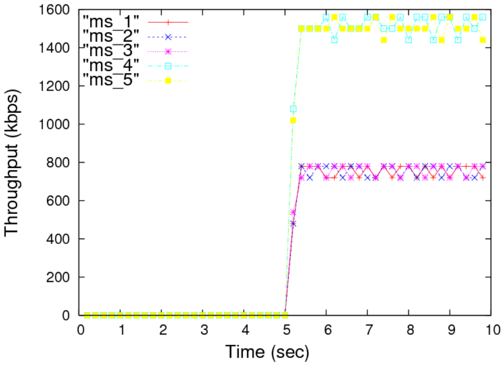

Fig. 6. Throughput of DRRF (MS_1 and MS_1 with traffic priority 2 and MS_3 to MS_5 with traffic priority 1)

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we investigate Deficit Round Robin (DRR) algorithm in the context of Mobile WiMAX networks. We compare DRR with a simple scheduling algorithm (imitating General Processor Sharing: GPS) and Deficit Round Robin with fragmentation enabled or DRRF.

To overcome the issue of varying link capacity, we use the slot allocation derived from the queue lengths and MCSs. The results show that all three scheduling algorithms can achieve perfect *throughput fairness but the total utilization of the link and the percentages of overhead are different.*

We compared the behavior of these three scheduling algorithms. In general, DRR results in frame underutilization and GPS results in high overhead due to fragmentation. In other words, DRRF can achieve high throughput with the least overhead because it allows fragmentation of the packets and the algorithm favors a full packet transmission.

Moreover, we extend DRRF algorithms to support users with minimum reserved traffic rate, maximum sustained traffic rate and traffic priority, the QoS parameters in Mobile WiMAX networks.

Although the simulations show the precise effect of each algorithm, more variations of simulation topologies, configurations, different traffic types including large number of mobile stations and variable packet sizes can be investigated. Moreover, with Automatic Repeat reQest (ARQ) and Hybrid ARQ features enabled, the scheduler needs to accommodate scheduling of the retransmission/feedback and the boundary of ARQ block [8]. One more requirement for Mobile WiMAX scheduling (without Hybrid ARQ) is that all downlink allocations be mapped to a rectangular area in the Orthogonal Frequency Division Multiple Access (OFDMA) frame. That restriction can reduce the throughput since some space may need to be left unused to make the allocation rectangular [9]. In addition, the optional packing feature can also be added. These issues need more investigation.

In our analysis we assume that the throughput is always non-zero, i.e., the worst case MCS is BPSK. However, it is possible that the channel conditions are so bad that even BPSK will not work then either the base station or the mobile station or both cannot transmit any bits for some time. The scheduler can simply record the transmitted opportunity of the MS during the unacceptable channel state condition. Then, once the mobile station is in an acceptable channel condition, that MS will get the opportunity back with some specified threshold distributed uniformly over some periods so as to prevent the starvation of other flows.

In a wireless network like mobile WiMAX, the throughput depends heavily upon the user location and whether the service providers should provide fairness among users located in the basement and those located in open outdoors is a debatable topic [9].

References

[1] IEEE P802.16Rev2/D2, “DRAFT Standard for Local and metropolitan area networks,” Part 16: Air Interface for Broadband Wireless Access Systems, Dec. 2007, 2094 pp.

[2] WiMAX Forum, “WiMAX System Evaluation Methodology V2.1,” Jul. 2008, 230 pp. URL=[http://www.wimaxforum.org/technology/documents]

[3] A. K. Parekh and R. G. Gallager, “A Generalized Processor Sharing approach to Flow control in Integrated Services Networks: The Single Node Case,” *IEEE/ACM Trans. Netw.*, 1993, vol. 1, no. 3, pp. 334-337.

[4] M. Shreedhar and G. Varghese, “Efficient fair queueing using deficit round robin,” in *Proc. ACM SIGCOMM Comm. Review.*, 1995, vol. 25, no. 4, pp. 231-242.

[5] UCB/LBNL/VINT, “Network Simulator – ns (version 2),” Jun. 2007. URL=[http://www.isi.edu/nsnam/ns/index.html]

[6] WiMAX Forum, “The Network Simulator NS-2 MAC+PHY Add-On for WiMAX,” Aug. 2007. URL=[http://www.wimaxforum.org]

[7] R. Jain, D.M. Chiu, and W. Hawe, “A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems,” in *DEC Research Report TR-301.*, 1984, 38 pp. URL=[http://www.cse.wustl.edu/~jain/papers/fairness.htm]

[8] A. Sayenko, O. Alanen, and T. Hamalainen, “ARQ aware scheduling for the IEEE 802.16 base station,” in *Proc. IEEE Computer Communication Conf.,* Beijing, China, 2008, pp. 2667-2673.

[9] C. So-In, R. Jain, and A. Al-Tamimi, “Scheduling in IEEE 802.16e WiMAX Networks: Key Issues and a Survey,” in *IEEE Journal on Selected Areas in Comm.,* 2008, vol. 27, no. 2, pp. 156-171.

Appendix

With Deficit Round Robin (DRR) scheduling algorithm, in wired networks, the quantum size in each round is recommended to be bigger than the maximum packets size such as 1500 bytes. Therefore, Tables VI and VII show the results with the same example discussed in Section II (4 MSs, 125 byte packet size and 300 byte frame capacity). In general, the quantum will be set to at least one packet size, 125.

For DRR and DRR with fragmentation awareness, after the quantum is derived by Max-Min fairness algorithm (see Section III.A), in case the quantum size is less than the number of slots required to send the full packet (we use 1500 bytes as the maximum packet size), we scale the quantum up proportionally. The simulation results show there is insignificant difference with the scaling.

TABLE VI

Updated deficit counter for DRR (Transmitted Packet Size /Deficit Counter)

Frame	1st		2nd	3rd		4th	5th	
Round	1			2			3	
MS_1	0/125	125/0	0/0	0/125	125/0	0/0	0/125	125/0
MS_2	0/125	125/0	0/0	0/125	125/0	0/0	0/125	125/0
MS_3	0/125	0/125	125/0	0/125	0/125	125/0	0/125	0/125
MS_4	0/125	0/125	125/0	0/125	0/125	125/0	0/125	0/125

TABLE VII

Updated deficit counter for DRRF (Transmitted Packet Size/Deficit Counter)

Frame	1st			2nd				3rd		4th		
Round	1				3							
MS_1	0/125	125/0	0/0	0/0	0/125	**100**/25	0/25	**25**/0	0/0	0/0	0/125	125/0
MS_2	0/125	125/0	0/0	0/0	0/125	0/125	0/125	125/0	0/0	0/0	0/125	**75**/50
MS_3	0/125	50/75	0/75	**75**/0	0/125	0/125	0/125	125/0	0/0	0/0	0/125	0/125
MS_4	0/125	0/125	0/125	125/0	0/125	0/125	0/125	**25**/100	0/100	**100**/0	0/125	0/125