MR-FQ: A Fair Scheduling Algorithm for Wireless Networks with Variable Transmission Rates

You-Chiun Wang
Yu-Chee Tseng
Department of Computer Science
National Chiao-Tung University
Hsin-Chu, 30010, Taiwan
yctseng@csie.nctu.edu.tw

Wen-Tsuen Chen
Department of Computer Science
National Tsing-Hua University
Hsin-Chu, 30043, Taiwan

Wireless networks are characterized by bursty and location-dependent errors. Although many fair scheduling algorithms have been proposed to address these issues, most of them assume a simple two-state channel model, where a channel can be either good or bad. In fact, the situation is not so pessimistic because different modulation techniques can be used to adapt to different channel conditions. Multirate transmission is a common technique for wireless networks nowadays. This leads to a dilemma: should fairness be built based on the amount of time that a user utilizes the medium or the amount of services that a user receives? In this work, the authors propose a *multirate wireless fair queueing* (MR-FQ) algorithm that allows a flow to transmit at different rates according to its channel condition and lagging degree. MR-FQ takes both time and service fairness into account. They demonstrate that MR-FQ can guarantee fairness and bounded delays for packet flows by mathematical modeling and analyses. Besides, simulation results show that MR-FQ can also increase the overall system throughput compared to other scheduling methods.

Keywords: Communication network, fair scheduling, multirate communication, quality of service (QoS), wireless network

1. Introduction

We have seen huge growth of wireless data services over the recent years. The increasing importance of real-time applications further demands provision of quality of service (QoS) and fair channel access among multiple packet flows over a shared, bandwidth-limited, error-prone wireless channel. In wireline networks, many fair scheduling algorithms [1-6] have been proposed to bound delays of packet transmission. However, wireless channels are characterized by the following features that distinguish themselves from wireline networks: (1) more serious bursty errors, (2) location-dependent errors, and (3) multirate communication capability. Bursty errors may break continuous services of a flow, while location-dependent errors may allow error-free flows to receive more services than they deserve, thus violating the fairness and bounded-delay requirements. A wireless channel may provide different transmission rates to different terminals depending on channel qualities (e.g., IEEE 802.11a supports 16 rates, while 802.11b supports 4 rates[1]). Due to these reasons, existing wireline solutions may not be suitable for the wireless networks [7-8].

Many fair scheduling algorithms have been proposed to address features (1) and (2) of wireless networks. In *idealized wireless fair queueing* (IWFQ) [9], each packet is associated with a *finish tag* computed by the principles of *weight fair queueing* (WFQ) [2], and the scheduler always serves the error-free packet with the smallest finish tag. When a flow suffers from errors, all its packets keep their original tags. After the flow exits from errors, its packets are likely to have smaller finish tags. So the scheduler will serve this flow and thus compensates its lost services. In *channel-condition independent fair queueing* (CIF-Q) [10], fairness is achieved by transferring the services allocated to error flows to those error-free flows. Then compensation services are dispatched to the former

SIMULATION, Vol. 81, Issue 8, August 2005 587-608

DOI: 10.1177/0037549705060239

1. IEEE 802.11a supports a set of data rates for 6, 9, 12, 18, $\cdots$, and 54 Mb/sec, whereas IEEE 802.11b supports a set of data rates for 1, 2, 5.5, and 11 Mb/sec.

proportional to their weights. In *server-based fairness approach* (SBFA) [11], a fraction of bandwidth is reserved particularly to compensate those error flows. A number of virtual servers, called *long-term fairness servers* (LTFS), are created for those flows that experienced errors. Later on, the reserved bandwidth is used to compensate these flows recorded in LTFS. *Wireless fair service* (WFS) [12] addresses the *delay-weight coupling problem* and alleviates the problem by assigning each flow with a *rate weight* and a *delay weight*. A flow is drained into the scheduler according to its rate weight but served according to its delay weight. In *traffic-dependent wireless fair queueing* (TD-FQ) [13], flows are separated into real-time flows and non-real-time flows. The scheduler gives higher priorities to real-time flows to reduce their queueing delays, while still maintaining fairness and bounded delays for all flows.

Unfortunately, feature (3) of wireless networks has not been well addressed in the area of fair queueing. Most works assume that a wireless channel is either in a *good (error-free)* state or a *bad (error)* state. Transmissions in a good state will succeed but completely fail in a bad state. In fact, the situation is not so pessimistic because different modulation techniques can be used to adapt to different channel conditions. The PHY of IEEE 802.11a/b are well-known examples, which can provide multirate transmission capabilities [14-15]. A simpler modulation (and thus a higher data rate) can be used when the signal-to-noise ratio (SNR) is sufficiently high, while a more complicated modulation (and thus a lower rate) can still be used under a bad channel [16]. Adopting multirate transmissions poses several challenges to fair queueing. First, there is a mismatch between the amount of service that a client receives and the amount of time that a server actually serves a client. To transmit the same amount of data, a client using a lower rate will take a longer time than one using a higher rate. So the concept of virtual time (such as finish tags) may need to be redefined. Second, when a flow that suffered from a bad channel exits from error, it may take a different amount of time for the system to compensate the flow, depending on its channel condition, thus making the design of compensation difficult. Third, the overall system performance may be degraded if there are too many low-rate flows.

In this work, we consider the fair scheduling problem in a wireless network with a TDMA medium access control (MAC) protocol and multirate communication capability. We propose a new algorithm called *multirate wireless fair queueing* (MR-FQ). MR-FQ can adjust a flow's transmission rate according to its channel condition and lagging degree. A flow is allowed to transmit at a lower rate to alleviate its lags only if it is lagging up to a certain degree. More specifically, the more serious a flow is lagging, the lower rate the flow is allowed to use. Such differentiation can take care of both fairness and system performance. Lower rate flows thus will not prolong other flows' delays. Besides, MR-FQ follows the idea in Wang, Ye, and Tseng [13] by separating real-time flows from non-real-time ones and compensates real-time lagging flows with higher priorities than non-real-time lagging flows to reduce the former flows' delays. However, such a special treatment does not starve non-real-time flows. Thus, MR-FQ can satisfy the delay-sensitive property of real-time applications while still maintaining fairness and bounded delays for all flows.

Several works have tried to differentiate flows' error conditions by adjusting their weights, but they still do not address the multirate feature. *Effort-limited fair* (ELF) [17] suggests adjusting each flow's weight in response to the error rate of that flow, up to a maximum defined by that flow's *power factor*. In *channel state independent wireless fair queueing* (CS-WFQ) [18], each flow i is associated with a fair share ϕ_i and a time-varying factor $f_i(t)$. The latter is used to adjust the former according to error rates. In *channel-adaptive fair queueing* (CAFQ) [19], the weight of each flow i is also adjusted by a factor $M(\Phi_i)^a$, where $M(\Phi_i)$ reflects the channel states and $0 \leq M(\Phi_i) \leq 1$. Certain works [20-23] address the multirate issue, but the focus is on assigning codes or adjusting transmission powers in code-division multiple access (CDMA) networks.

The remainder of this article is organized as follows. Section 2 presents our MR-FQ algorithm. In section 3, we demonstrate the properties of MR-FQ (such as fairness and bounded delays) by mathematical modeling and analyses. Section 4 presents some simulation results to verify the effectiveness and properties of MR-FQ. Conclusions are drawn in section 5.

2. The MR-FQ Algorithm

2.1 System Model

We consider a base station (BS) as in Figure 1. Packets arriving at the BS are classified into real-time traffic and non-real-time traffic and dispatched into different flow queues depending on their destination mobile stations. These traffic flows are sent to the *MR-FQ packet scheduler*, which is responsible for scheduling flows and transmitting the head-of-line (HOL) packet of the selected flow to the *MAC and transmission* (MT) module. The MT module can transmit at n rates $\hat{C}_1, \hat{C}_2, \cdots$, and $\hat{C}_n$, where $\hat{C}_1 > \hat{C}_2 > \cdots > \hat{C}_n$. It also measures the current channel condition to each mobile station and determines the most appropriate rate to communicate with the station (several works [16, 24-26] have addressed the rate selection problem, but this is out of the scope of this work). The information of the best rate is also reported to the scheduler for making a decision. For simplicity, we assume that the BS has immediate knowledge of the best rate for each station. Note that this also includes the worst case where the channel is too bad to be used, in which case we can regard the best rate to be zero.

2.2 Service Fairness vs. Time Fairness

With the emergence of multirate communication, the concept of fairness may be defined in two ways. One is *service fairness*, which means that the difference between services

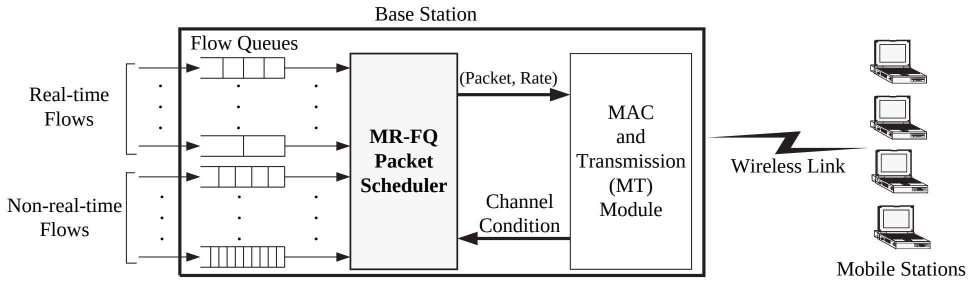


Figure 1. System architecture of multirate wireless fair queueing (MR-FQ)

received by any two flows should be bounded, and the other is *time fairness*, which means that the difference between the amounts of transmission time of any two flows should be bounded. Formally, let w_i be the weight of flow i, and $\Phi_i^s(t_1, t_2)$ and $\Phi_i^t(t_1, t_2)$ be the amount of services and the amount of time that flow i receives/uses during the time interval $[t_1, t_2)$, respectively. Then, for any two flows i and j, during any $[t_1, t_2)$,

$$\left| \frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j} \right| \leq \sigma_s \tag{1}$$

holds if service fairness is desired, and

$$\left| \frac{\Phi_i^t(t_1, t_2)}{w_i} - \frac{\Phi_j^t(t_1, t_2)}{w_j} \right| \leq \sigma_t \tag{2}$$

holds if time fairness is desired, where σ_s and σ_t are small, nonnegative numbers.

We observe that in a single-rate environment, equations (1) and (2) are equivalent. However, in a multirate environment, equations (1) and (2) may not be satisfied at the same time. If service fairness is desired, then flows using lower rates will occupy more of the medium time. On the contrary, if time fairness is desired, then flows using higher rates will transmit more data. The concept is illustrated in Figure 2. Furthermore, when the rates used by stations exhibit higher variation, the trade-off between service and time fairness is more significant (solid line in Fig. 2). When the variation is lower, the trade-off is less significant (dashed line in Fig. 2). When the variation is 0, this degenerates to the single-rate case (thick line in Fig. 2).

2.3 Scheduling Policy

Figure 2 leads to the following guidelines in the design of MR-FQ. First, the concept of virtual time is redefined based on the concept of time fairness. However, we differentiate flows according to their lagging degrees. A flow is allowed to use a lower transmission rate only if it is suffering from a higher lagging degree. In this way, we can take care of service fairness. So the system performance would not be hurt when there exist too many low-rate stations.

In MR-FQ, like traditional fair queueing works, each flow i is assigned a *weight* w_i to represent the ideal fraction of bandwidth that the system commits to it. For each flow i, we maintain a *virtual time* v_i to record the nominal services received by it and a *lagging index* lag_i to record its credits/debts. The former is used to compete with other flows for services, while the latter is used to arrange compensation services. The actual normalized service received by flow i is $v_i - lag_i/w_i$. Flow i is called *leading* if $lag_i < 0$, called *lagging* if $lag_i > 0$, and called *satisfied* if $lag_i = 0$. Furthermore, depending on its queue content, a flow is called *backlogged* if its queue is nonempty, called *nonbacklogged* if its queue is empty, and called *active* if it is backlogged or nonbacklogged but leading. Note that MR-FQ only selects active flows to serve. When a nonbacklogged but leading flow (i.e., an active flow) is chosen, its service will actually be transferred to another flow for compensation purpose. Besides, whenever a flow i transits from nonbacklogged to backlogged, its virtual time v_i is set to $max\{v_i, min_{j \in A}\{v_j\}\}$, where A is the set of all active flows.

Figure 3 outlines the scheduling policy of MR-FQ. First, the active flow i with the smallest virtual time v_i is selected. If flow i is backlogged, the *rate selection scheme* is called to compute the best rate r to transmit for flow i. If the result is $r \leq 0$, that means either flow i has a bad channel condition or its current lagging degree does not allow it to transmit (refer to section 2.3.1 for details). Otherwise, if flow i is nonleading, the HOL packet of flow i will be served. Then we update the virtual time of flow i as follows:

$$v_i = v_i + \left(\frac{l_p}{w_i} \times \frac{\hat{C}_1}{r} \right), \tag{3}$$

where l_p is the length of the packet. Note that the ratio $\frac{\hat{C}_1}{r}$ is to reflect the concept of time fairness. The amount of increase in v_i is inverse to the transmission rate r. So if a lower r is used, the less competitive flow i will be in the next round.

If flow i is overserved (i.e., leading), the *graceful degradation scheme* is activated to check if flow i is still eligible for the service (refer to section 2.3.2). In case that flow i

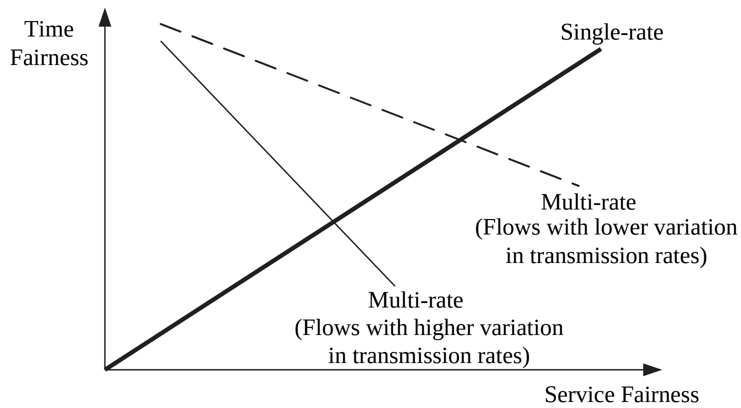


Figure 2. The trade-off between service fairness and time fairness

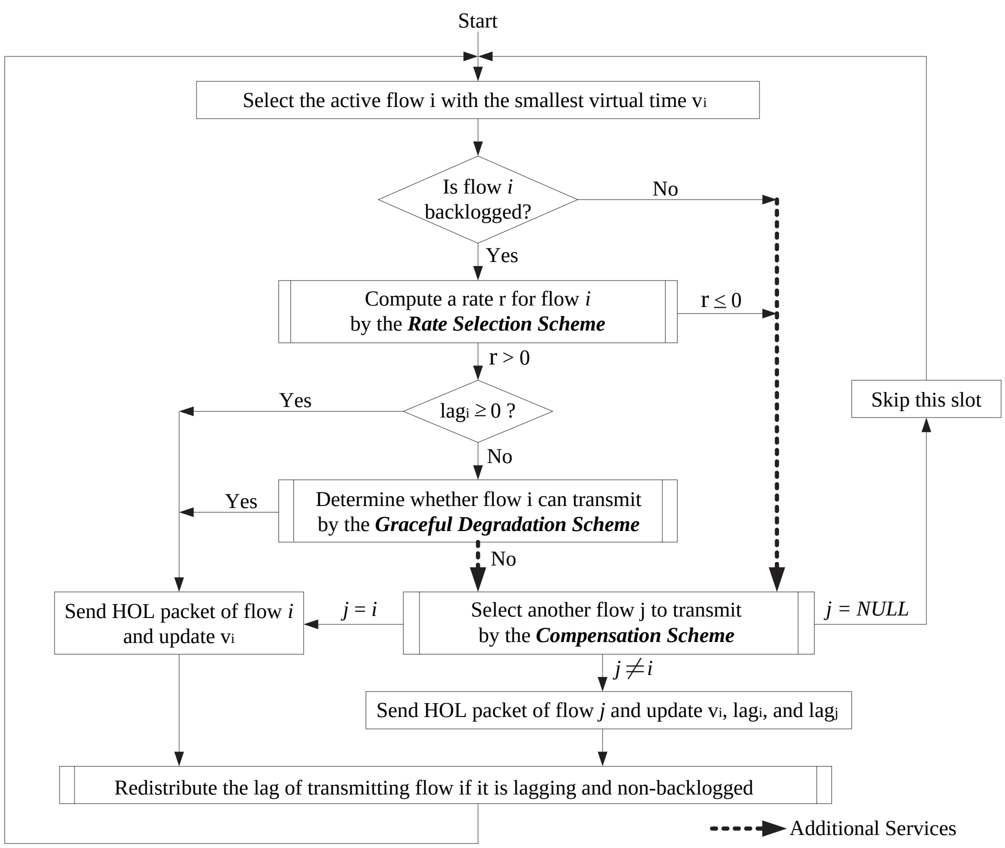


Figure 3. The multirate wireless fair queueing (MR-FQ) algorithm

has to give up its service due to an empty queue, a bad channel condition, or a rejection decision by the graceful degradation scheme, the service is transferred to the *compensation scheme* to select another flow j to serve (refer to section 2.3.3). If the scheme fails to select any flow, this service is just wasted. If the scheme still selects flow i to serve, then we send its HOL packet and update v_i according to equation (3). If another flow j ($\neq i$) is selected, flow j's packet is sent, and the values of v_i, lag_i, and lag_j are updated as follows:

$$v_i = v_i + l_{p'}/w_i, \quad (4)$$

$$lag_i = lag_i + l_{p'}, \quad (5)$$

$$lag_j = lag_j - l_{p'}, \quad (6)$$

where p' is the packet being sent. Note that in this case, we "charge" to flow i by increasing its virtual time (i.e., equation (4)) but "credit" to lag_i of flow i (i.e., equation (5)) and "debit" to lag_j of flow j (i.e., equation (6)). Since flow i is not actually served, equation (4) is equivalent to equation (3) with $r = \hat{C}_1$.

Whenever the scheduler serves any flow i, it has to check the queue size of flow i. If flow i's queue state changes to nonbacklogged and it is still lagging, we distribute its credit to other flows that are in debt and reset its credit to zero. This is because the flow does not need the credit any more [27]. We give flow i's credit to other flows in debt proportional to their weights; that is, for each flow k such that $lag_k < 0$, we set

$$lag_k = lag_k + lag_i \times \frac{r_k}{\sum_{lag_m<0} r_m}.$$

Then we reset $lag_i = 0$.

Below, we introduce the three schemes: the rate selection scheme, the graceful degradation scheme, and the compensation scheme. Table 1 summarizes symbols used in MR-FQ.

2.3.1 Rate Selection Scheme

When a backlogged flow i is selected, the rate selection scheme is invoked to choose a suitable transmission rate for flow i according to its lagging degree and channel condition. The basic idea is to permit different ranges of transmission rates according to flow i's normalized lag, $\frac{lag_i}{w_i}$. To help a seriously lagging flow to alleviate its huge lag, we allow it to use a larger range of rates. Specifically, we set up $n-1$ levels of lagging thresholds $\delta_1, \delta_2, \cdots, \delta_{n-1}$. A flow with a normalized lag exceeding δ_i is allowed to use a rate as low as $\hat{C}_{i+1}$, $i \leq n-1$. Figure 4 shows the mapping of lagging degrees to allowable transmission rates. If flow i's current best rate falls within the allowable range, the rate is returned. Otherwise, a negative value is returned to indicate a failure. For example, if flow i satisfies $\delta_2 < \frac{lag_i}{w_i} \leq \delta_3$ and its current best rate is $\hat{C}_2$, then $\hat{C}_2$ is returned. If the current best rate is $\hat{C}_5$, then a negative value is returned.

2.3.2 Graceful Degradation Scheme

When a leading flow i is selected for service, the graceful degradation scheme is triggered to check its leading amount. A leading flow is allowed to receive an amount of additional service proportional to its normal services. Specifically, when a flow i transits from lagging/satisfied to leading, we set up a parameter $s_i = \alpha \cdot v_i$, where α ($0 \leq \alpha \leq 1$) is a system-defined constant. Later on, flow i's virtual time is increased each time when it is selected by the scheduler (according to earlier discussion, "selected" does not mean that it is actually "served"). Let v_i' be flow i's current virtual time when it is selected. We allow flow i to be served if $s_i \leq \alpha v_i'$. If so, s_i is updated as $s_i + l_p/r_i$, where l_p is the length of the packet. Intuitively, flow i can enjoy approximately $\alpha(v_i' - v_i)$ services when it is leading.

Moreover, to distinguish real-time from non-real-time flows, we substitute the above α by a parameter α_R for real-time flows and by α_N for non-real-time flows. We set $\alpha_R > \alpha_N$ to distinguish their priorities.

2.3.3 Compensation Scheme

When the selected flow i does not have a satisfactory channel condition or fails to pass the graceful degradation scheme, the compensation scheme is triggered (reflected by *additional services* in Fig. 3). Figure 5 shows how to dispatch additional services. Flows are prioritized according to the following rules. First, lagging flows have a higher priority over nonlagging flows to receive such services. Second, flows that can use higher rates to transmit have a higher priority over flows that can use lower rates. Third, among lagging flows of the same best rate, real-time flows and non-real-time ones will share the services according to some ratio. Note that the third rule is not applied to leading flows because such flows suffer no lagging.

Next, we elaborate on the third rule. When dispatching additional services to lagging flows (i.e., flows on the left-hand side in Fig. 5), we keep track of the services received by real-time ones and non-real-time ones. Let $L_R = L_R^1 \cup L_R^2 \cup \cdots \cup L_R^n$ be the set of real-time lagging flows and $L_N = L_N^1 \cup L_N^2 \cup \cdots \cup L_N^n$ the set of non-real-time lagging flows. To let real-time lagging flows receive more fraction of additional services without starving non-real-time lagging flows, we assign weights W_R and W_N (system parameters) to L_R and L_N, respectively, to control the fractions of additional services they already received, where $W_R > W_N$. A virtual time V_R (respectively, V_N) is used to record the normalized additional services received by L_R (respectively, L_N). Flows in Figure 5 are checked from left to right. When both L_R^k and L_N^k are nonempty, $1 \leq k \leq n$, the service is given to L_R if $V_R \leq V_N$ and to L_N otherwise. When only one of L_R^k and L_N^k is nonempty, the service is given to that one, independent of the values of V_R and V_N. When a flow in L_R receives the service, V_R is updated as

Table 1. Summary of symbols used in multirate wireless fair queueing (MR-FQ)

Symbols	Definition
$\hat{C}_1, \hat{C}_2, \cdots, \hat{C}_n$	All transmission rates
w_i	Weight of flow i
v_i	Virtual time of flow i
lag_i	Credits/debts of flow i
$\delta_1, \delta_2, \cdots, \delta_{n-1}$	Thresholds to distinguish lagging degrees of flows
s_i	Graceful degradation service index of flow i when $lag_i < 0$
α_R, α_N	Graceful degradation ratios for real-time and non-real-time flows
L_R, L_N	Real-time lagging flows and non-real-time lagging flows
W_R, W_N	Weights of L_R and L_N, respectively
V_R, V_N	Normalized amounts of additional services received by L_R and L_N, respectively
B	Bound of differences of services for L_R and L_N
c_i	Normalized amounts of additional services received by flow i when $lag_i > 0$
f_i	Normalized amount of additional services received by flow i when $lag_i \leq 0$

Lagging degrees	$\hat{C}_1$	$\hat{C}_2$	$\hat{C}_3$	$\cdots$	$\hat{C}_{n-2}$	$\hat{C}_{n-1}$	$\hat{C}_n$
$\frac{lag_i}{w_i} \leq \delta_1$	√						
$\delta_1 < \frac{lag_i}{w_i} \leq \delta_2$	√	√					
$\delta_2 < \frac{lag_i}{w_i} \leq \delta_3$	√	√	√				
⋮				⋮			
$\delta_{n-3} < \frac{lag_i}{w_i} \leq \delta_{n-2}$	√	√	√	$\cdots$	√		
$\delta_{n-2} < \frac{lag_i}{w_i} \leq \delta_{n-1}$	√	√	√	$\cdots$	√	√	
$\delta_{n-1} \leq \frac{lag_i}{w_i}$	√	√	√	$\cdots$	√	√	√

Figure 4. The mapping of lagging degrees to allowable transmission rates (indicated by checkmarks) in the rate selection scheme

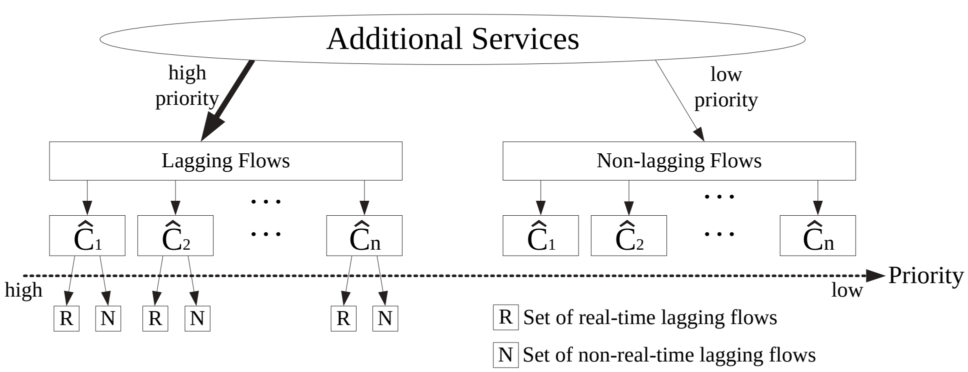


Figure 5. Dispatching additional services in the compensation scheme

$$V_R = min\left\{V_R + \frac{l_p}{W_R}, \frac{B + V_N W_N}{W_R}\right\}, \quad (7)$$

where l_p is the length of the packet being transmitted, and B is a predefined value to bound the difference between V_R and V_N. Similarly, when a flow in L_N receives the service, V_N is updated as

$$V_N = min\left\{V_N + \frac{l_p}{W_N}, \frac{B + V_R W_R}{W_N}\right\}. \quad (8)$$

Note that to avoid $V_R \gg V_N$ (respectively, $V_N \gg V_R$), which may cause flows in L_R (respectively, L_N) to starve, we set up a bound $|V_R W_R - V_N W_N| \leq B$. This is reflected by the second term in the right-hand side of equations (7) and (8).

When the scheduler selects either L_R^k or L_N^k, it distributes additional services proportional to the weights of flows in that set. Specifically, for each flow i, we maintain a *compensation virtual time* c_i to keep track of the normalized amount of additional services received by flow i. The scheduler selects the flow i with the smallest c_i to serve and then updates c_i as

$$c_i = c_i + \left(\frac{l_p}{w_i} \times \frac{\hat{C}_1}{\hat{C}_k}\right). \quad (9)$$

Initially, when a flow i newly enters L_R or L_N, its c_i is set to

$$c_i = max\{c_i, min\{c_j \mid \text{flow } j \text{ belongs to the same set of flow } i\ (L_R \text{ or } L_N),\ j \neq i\}\}.$$

If there is no lagging flow in the previous stage, the service is returned back to the originally selected flow if it is a leading flow but rejected by the graceful degradation scheme. Otherwise, the service is given to a nonlagging flow that can use the highest rate. In case of a tie, MR-FQ dispatches the services proportional to some weights. Specifically, each flow i is assigned with an *extra virtual time* f_i to keep track of the normalized amount of additional services received by flow i when it is nonlagging ($lag_i \leq 0$). Whenever a backlogged flow i that can send becomes nonlagging, f_i is set to

$$f_i = max\{f_i, min\{f_j \mid \text{flow } j \text{ is backlogged, nonlagging, and can send},\ j \neq i\}\}.$$

The scheduler selects the flow i with the smallest f_i to serve. When flow i receives the service, f_i is updated as

$$f_i = f_i + \left(\frac{l_p}{w_i} \times \frac{\hat{C}_1}{r}\right), \quad (10)$$

where r is the current best rate for flow i.

3. Fairness and Delay Analyses

In this section, we demonstrate that MR-FQ can guarantee fairness (including service fairness and time fairness) and bounded delays for packet flows by mathematical modeling and analyses. Our analyses rely on the following assumptions: (1) $\alpha_R > \alpha_N$, (2) $W_R > W_N$, (3) $B > \hat{L}_{max}$, and (4) $r_i \in \{\hat{C}_1, \cdots, \hat{C}_n\}$, where $\hat{L}_{max}$ is the maximum length of a packet and r_i is the transmission rate used by flow i. A flow is called *allowed-to-send* if the rate selection scheme returns a positive transmission rate to it, and it is called a *candidate* if it can use a higher rate compared to other flows such that the scheduler may choose it to receive additional services in the compensation scheme. Besides, we let r_i^{min} be the smallest transmission rate that flow i has ever used during the nearest time interval when flow i is active. The lemmas used in the proofs are contained in the appendix.

3.1 Service Fairness

Theorems 1 and 2 show the service fairness guaranteed by MR-FQ under some constraints. Theorem 1 is for flows that have the similar conditions, and theorem 2 provides some bounds on differences of services received by L_R and L_N.

THEOREM 1. For any two active flows i and j, assume that both flows are continuously backlogged and allowed-to-send and remain in the same state (leading, lagging, or satisfied) during a time interval $[t_1, t_2)$. Let r_{RSC} and r_{CS} be the transmission rates used by these flows in the rate selection scheme and the compensation scheme during $[t_1, t_2)$, respectively, where r_{RSC} and r_{CS} are both in $\{\hat{C}_1, \cdots, \hat{C}_n\}$, and their values do not change during $[t_1, t_2)$. Then the difference between the normalized services received by flows i and j during $[t_1, t_2)$ satisfies the following inequality:

$$\left|\frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j}\right| \leq \beta \cdot \frac{\hat{L}_{max}}{w_i} + \gamma \cdot \frac{\hat{L}_{max}}{w_j},$$

where $\Phi_i^s(t_1, t_2)$ represents the services received by flow i during $[t_1, t_2)$, and

$$(\beta, \gamma) = \begin{cases} \left(\frac{r_{RSC}}{r_i^{min}} + 1, \frac{r_{RSC}}{r_j^{min}} + 1\right), \\ \text{if both flows are lagging but not candidates} \\ \left(\frac{r_{RSC}+r_{CS}}{r_i^{min}} + 1, \frac{r_{RSC}+r_{CS}}{r_j^{min}} + 1\right), \\ \text{if both flows are lagging and candidates} \\ \left(\frac{r_{RSC}+r_{CS}}{r_i^{min}} + 1, \frac{r_{RSC}+r_{CS}}{r_j^{min}} + 1\right), \\ \text{if both flows are satisfied} \\ \left(\frac{r_{CS}+\alpha_R\hat{C}_1}{r_i^{min}} + 2, \frac{r_{CS}+\alpha_R\hat{C}_1}{r_j^{min}} + 2\right), \\ \text{if both flows are real-time leading flows} \\ \left(\frac{r_{CS}+\alpha_N\hat{C}_1}{r_i^{min}} + 2, \frac{r_{CS}+\alpha_N\hat{C}_1}{r_j^{min}} + 2\right), \\ \text{if both flows are non-real-time leading flows} \\ \left(\frac{r_{CS}}{r_i^{min}} + 2, \frac{r_{CS}+2\alpha_N\hat{C}_1}{r_j^{min}} + 2\right), \\ \text{if } i \text{ and } j \text{ are real-time and non-real-time leading flows, respectively} \end{cases}.$$

Proof. A lagging flow that is allowed-to-send is not necessarily a candidate since there may exist other lagging flows that can use higher rates to transmit. Thus, we have to consider the five cases: (1) flows i and j are both lagging but not candidates, (2) flows i and j are both lagging and candidates, (3) flows i and j are both satisfied, (4) flows i and j are both leading and have the same traffic type, and (5) flow i is a real-time leading flow and flow j is a non-real-time leading flow during the entire time interval $[t_1, t_2)$.

Case (1): In this case, any flow i that is lagging but not a candidate can only receive services each time when it is selected by v_i. Since v_i is updated *before* a packet is transmitted, the services received by flow i may deviate from its virtual time by one packet. Besides, the services received by flow i is $v_i \times \frac{r_{RSC}}{\hat{C}_1}$. Thus, we have

$$\frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) - \frac{\hat{L}_{max}}{w_i} \leq \frac{\Phi_i^s(t_1, t_2)}{w_i} \leq \frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) + \frac{\hat{L}_{max}}{w_i}. \quad (11)$$

Applying equation (11) to flows i and j, we have

$$\frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) - \frac{\hat{L}_{max}}{w_i} - \left(\frac{r_{RSC}}{\hat{C}_1}(v_j(t_2) - v_j(t_1)) + \frac{\hat{L}_{max}}{w_j}\right) \leq \frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j} \leq \frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) + \frac{\hat{L}_{max}}{w_i} - \left(\frac{r_{RSC}}{\hat{C}_1}(v_j(t_2) - v_j(t_1)) - \frac{\hat{L}_{max}}{w_j}\right).$$

By lemma 1, the leftmost term can be reduced to

$$\frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_j(t_2) - (v_i(t_1) - v_j(t_1))) - \left(\frac{\hat{L}_{max}}{w_i} + \frac{\hat{L}_{max}}{w_j}\right) \geq -\left(\frac{r_{RSC}}{r_i^{min}} + 1\right)\frac{\hat{L}_{max}}{w_i} - \left(\frac{r_{RSC}}{r_j^{min}} + 1\right)\frac{\hat{L}_{max}}{w_j}.$$

Similarly, the rightmost term would be less than or equal to $\left(\frac{r_{RSC}}{r_i^{min}} + 1\right)\frac{\hat{L}_{max}}{w_i} + \left(\frac{r_{RSC}}{r_j^{min}} + 1\right)\frac{\hat{L}_{max}}{w_j}$, so

$$\left|\frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j}\right| \leq \left(\frac{r_{RSC}}{r_i^{min}} + 1\right)\frac{\hat{L}_{max}}{w_i} + \left(\frac{r_{RSC}}{r_j^{min}} + 1\right)\frac{\hat{L}_{max}}{w_j}.$$

Case (2): In this case, both flows can receive services each time when they are selected by v_i/v_j or receive additional services from others by c_i/c_j. Since the additional services received by flow i are $c_i \times \frac{r_{CS}}{\hat{C}_1}$, we have

$$\frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) + \frac{r_{CS}}{\hat{C}_1}(c_i(t_2) - c_i(t_1)) - \frac{\hat{L}_{max}}{w_i} \leq \frac{\Phi_i^s(t_1, t_2)}{w_i} \leq \frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) + \frac{r_{CS}}{\hat{C}_1}(c_i(t_2) - c_i(t_1)) + \frac{\hat{L}_{max}}{w_i}.$$

Similarly to case 1, by lemmas 1 and 2, we can obtain

$$\left|\frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j}\right| \leq \left(\frac{r_{RSC} + r_{CS}}{r_i^{min}} + 1\right)\frac{\hat{L}_{max}}{w_i} + \left(\frac{r_{RSC} + r_{CS}}{r_j^{min}} + 1\right)\frac{\hat{L}_{max}}{w_j}.$$

Case (3): In this case, both flows can receive services each time when they are selected by v_i/v_j or when they receive additional services from another flow by f_i/f_j. Besides, since the additional services received by flow i are $f_i \times \frac{r_{CS}}{\hat{C}_1}$, we have

$$\frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) + \frac{r_{CS}}{\hat{C}_1}(f_i(t_2) - f_i(t_1)) - \frac{\hat{L}_{max}}{w_i} \leq \frac{\Phi_i^s(t_1, t_2)}{w_i} \leq \frac{r_{RSC}}{\hat{C}_1}(v_i(t_2) - v_i(t_1)) + \frac{r_{CS}}{\hat{C}_1}(f_i(t_2) - f_i(t_1)) + \frac{\hat{L}_{max}}{w_i}.$$

Consequently, similar to case 1, by lemmas 1 and 3, we can obtain

$$\left|\frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j}\right| \leq \left(\frac{r_{RSC} + r_{CS}}{r_i^{min}} + 1\right)\frac{\hat{L}_{max}}{w_i} + \left(\frac{r_{RSC} + r_{CS}}{r_j^{min}} + 1\right)\frac{\hat{L}_{max}}{w_j}.$$

Case (4): An allowed-to-send, backlogged, leading flow i can receive services by s_i and additional services from other flows by f_i. So the total services received by flow i during $[t_1, t_2)$ are bounded as

$$s_i(t_2) - s_i(t_1) + \frac{r_{CS}}{\hat{C}_1}(f_i(t_2) - f_i(t_1)) - \frac{\hat{L}_{max}}{w_i} \leq \frac{\Phi_i(t_1, t_2)}{w_i}$$
$$\leq s_i(t_2) - s_i(t_1) + \frac{r_{CS}}{\hat{C}_1}(f_i(t_2) - f_i(t_1)) + \frac{\hat{L}_{max}}{w_i}.$$

Applying the previous inequality to flows i and j, we have

$$\frac{r_{CS}}{\hat{C}_1}(f_i(t_2) - f_j(t_2) - f_i(t_1) + f_j(t_1)) + s_i(t_2)$$
$$- s_j(t_2) - s_i(t_1) + s_j(t_1) - \frac{\hat{L}_{max}}{w_i} - \frac{\hat{L}_{max}}{w_j}$$
$$\leq \frac{\Phi_i^s(t_1, t_2)}{w_i} - \frac{\Phi_j^s(t_1, t_2)}{w_j} \leq \frac{r_{CS}}{\hat{C}_1}(f_i(t_2)$$
$$- f_j(t_2) - f_i(t_1) + f_j(t_1)) + s_i(t_2) - s_j(t_2)$$
$$- s_i(t_1) + s_j(t_1) + \frac{\hat{L}_{max}}{w_i} + \frac{\hat{L}_{max}}{w_j}. \quad (12)$$

Applying lemma 4 twice to flows i and j and subtracting one by the other, we have

$$\alpha\left(v_i(t) - v_j(t)\right) + \alpha\left(\frac{\hat{L}_{max}}{w_j} - \frac{\hat{L}_{max}}{w_i}\right) - \frac{\hat{L}_{max}}{w_j}$$
$$\leq s_i(t) - s_j(t) \leq \alpha\left(v_i(t) - v_j(t)\right)$$
$$+ \alpha\left(\frac{\hat{L}_{max}}{w_j} - \frac{\hat{L}_{max}}{w_i}\right) + \frac{\hat{L}_{max}}{w_i}.$$

By lemma 1, we can rewrite the inequality as

$$-\left(\alpha\frac{\hat{C}_1}{r_j^{min}} - \alpha + 1\right)\frac{\hat{L}_{max}}{w_j} - \alpha\frac{\hat{L}_{max}}{w_i} \leq s_i(t)$$
$$- s_j(t) \leq \left(\alpha\frac{\hat{C}_1}{r_i^{min}} - \alpha + 1\right)\frac{\hat{L}_{max}}{w_i} + \alpha\frac{\hat{L}_{max}}{w_j}. \quad (13)$$

Applying equation (13) and lemma 3 to equation (12), we have

$$\left|\frac{\Phi_i(t_1, t_2)}{w_i} - \frac{\Phi_j(t_1, t_2)}{w_j}\right| \leq \left(\frac{r_{CS} + \alpha\hat{C}_1}{r_i^{min}} + 2\right)\frac{\hat{L}_{max}}{w_i}$$
$$+ \left(\frac{r_{CS} + \alpha\hat{C}_1}{r_j^{min}} + 2\right)\frac{\hat{L}_{max}}{w_j},$$

where $\alpha = \alpha_R$ if these are real-time flows, and $\alpha = \alpha_N$ if they are non-real-time flows.

Case (5): Applying lemma 4 to flows i and j and taking a subtraction leads to

$$\alpha_R v_i(t) - \alpha_R\frac{\hat{L}_{max}}{w_i} - \left(\alpha_N v_j(t) - (\alpha_N - 1)\frac{\hat{L}_{max}}{w_j}\right)$$
$$\leq s_i(t) - s_j(t) \leq \alpha_R v_i(t) - (\alpha_R - 1)\frac{\hat{L}_{max}}{w_i}$$
$$- \left(\alpha_N v_j(t) - \alpha_N\frac{\hat{L}_{max}}{w_j}\right) = S_{right}. \quad (14)$$

By lemma 1 and the $\alpha_R > \alpha_N$ principle, the left-hand side of equation (14) becomes

$$\alpha_R v_i(t) - \alpha_N v_j(t) + \alpha_N\frac{\hat{L}_{max}}{w_j} - \alpha_R\frac{\hat{L}_{max}}{w_i} - \frac{\hat{L}_{max}}{w_j}$$
$$> \alpha_N(v_i(t) - v_j(t)) + \alpha_N\frac{\hat{L}_{max}}{w_j} - \alpha_R\frac{\hat{L}_{max}}{w_i} - \frac{\hat{L}_{max}}{w_j}$$
$$\geq -\alpha_R\frac{\hat{L}_{max}}{w_i} - \left(\alpha_N\frac{\hat{C}_1}{r_j^{min}} - \alpha_N + 1\right)\frac{\hat{L}_{max}}{w_j}.$$

Consider the right-hand side of equation (14). There are two cases for the term $\alpha_R v_i(t) - \alpha_N v_j(t)$. If $\alpha_R v_i(t) - \alpha_N v_j(t) \geq 0$, we have $v_i(t) \geq \frac{\alpha_N}{\alpha_R} v_j(t)$. By lemma 1,

$$S_{right} \leq \alpha_N\left(v_j(t) - v_i(t)\right) + \alpha_N\frac{\hat{L}_{max}}{w_j} - \alpha_R\frac{\hat{L}_{max}}{w_i} + \frac{\hat{L}_{max}}{w_i}$$
$$\leq \left(\alpha_N\frac{\hat{C}_1}{r_j^{min}} + \alpha_N\right)\frac{\hat{L}_{max}}{w_j} + (1 - \alpha_R)\frac{\hat{L}_{max}}{w_i}.$$

If $\alpha_R v_i(t) - \alpha_N v_j(t) < 0$, we have

$$S_{right} \leq \alpha_N\frac{\hat{L}_{max}}{w_j} + (1 - \alpha_R)\frac{\hat{L}_{max}}{w_i}.$$

These two cases together imply $S_{right} \leq \left(\alpha_N\frac{\hat{C}_1}{r_j^{min}} + \alpha_N\right)\frac{\hat{L}_{max}}{w_j} + (1 - \alpha_R)\frac{\hat{L}_{max}}{w_i}$. So we have

$$-\alpha_R\frac{\hat{L}_{max}}{w_i} - \left(\alpha_N\frac{\hat{C}_1}{r_j^{min}} - \alpha_N + 1\right)\frac{\hat{L}_{max}}{w_j} \leq s_i(t)$$
$$- s_j(t) \leq \left(\alpha_N\frac{\hat{C}_1}{r_j^{min}} + \alpha_N\right)\frac{\hat{L}_{max}}{w_j} + (1 - \alpha_R)\frac{\hat{L}_{max}}{w_i}. \quad (15)$$

By applying equation (15) and lemma 3 to equation (12), we have

$$\left|\frac{\Phi_i^s(t_1,t_2)}{w_i}-\frac{\Phi_j^s(t_1,t_2)}{w_j}\right| \le \left(\frac{r_{CS}}{r_i^{min}}+2\right)\frac{\hat{L}_{max}}{w_i} + \left(\frac{r_{CS}+2\alpha_N\hat{C}_1}{r_j^{min}}+2\right)\frac{\hat{L}_{max}}{w_j}.$$

THEOREM 2. The difference between normalized additional services received by L_R and L_N in any time interval $[t_1, t_2)$ during which both sets remain active (i.e., there exists at least one candidate in each set) satisfies the following inequality:

$$\left|\frac{\Phi_R(t_1,t_2)}{W_R}-\frac{\Phi_N(t_1,t_2)}{W_N}\right| \le \frac{B+\hat{L}_{max}}{W_R}+\frac{B+\hat{L}_{max}}{W_N},$$

where $\Phi_R(t_1, t_2)$ and $\Phi_N(t_1, t_2)$ are additional services received by L_R and L_N during $[t_1, t_2)$, respectively.

Proof. Since V_R is updated *before* a packet is transmitted, it follows that the total additional services received by L_R during $[t_1, t_2)$ are bounded by

$$V_R(t_2)-V_R(t_1)-\frac{\hat{L}_{max}}{W_R} \le \frac{\Phi_R(t_1,t_2)}{W_R} \le V_R(t_2) - V_R(t_1)+\frac{\hat{L}_{max}}{W_R}.$$

Similarly, for V_N, we have

$$V_N(t_2)-V_N(t_1)-\frac{\hat{L}_{max}}{W_N} \le \frac{\Phi_N(t_1,t_2)}{W_N} \le V_N(t_2) - V_N(t_1)+\frac{\hat{L}_{max}}{W_N}.$$

Therefore, we have

$$V_R(t_2)-V_R(t_1)-\frac{\hat{L}_{max}}{W_R}-\left(V_N(t_2)-V_N(t_1)+\frac{\hat{L}_{max}}{W_N}\right) \le \frac{\Phi_R(t_1,t_2)}{W_R}-\frac{\Phi_N(t_1,t_2)}{W_N} \le V_R(t_2)-V_R(t_1) + \frac{\hat{L}_{max}}{W_R}-\left(V_N(t_2)-V_N(t_1)-\frac{\hat{L}_{max}}{W_N}\right).$$

By lemma 5, we can rewrite the inequality as

$$-\left(\frac{B+\hat{L}_{max}}{W_R}+\frac{B+\hat{L}_{max}}{W_N}\right) \le \frac{\Phi_R(t_1,t_2)}{W_R} - \frac{\Phi_N(t_1,t_2)}{W_N} \le \frac{B+\hat{L}_{max}}{W_R}+\frac{B+\hat{L}_{max}}{W_N}$$

$$\Rightarrow \left|\frac{\Phi_R(t_1,t_2)}{W_R}-\frac{\Phi_N(t_1,t_2)}{W_N}\right| \le \frac{B+\hat{L}_{max}}{W_R} + \frac{B+\hat{L}_{max}}{W_N}.$$

3.2 Time Fairness

Theorem 3 shows the time fairness guaranteed by MR-FQ. Since v_i, c_i, and f_i reflect the transmission time used by flow i, the proof of theorem 3 is similar to that of theorem 1, except that we do not multiply v_i, c_i, and f_i by $\frac{r_{RSC}}{\hat{C}_1}$ or $\frac{r_{CS}}{\hat{C}_1}$ factors. Thus, we omit the proof of theorem 3.

THEOREM 3. For any two active flows i and j, the difference between the normalized transmission time used by flows i and j in any time interval $[t_1, t_2)$ during which both flows are continuously backlogged and allowed-to-send and remain in the same state (leading, lagging, or satisfied) satisfies the following inequality:

$$\left|\frac{\Phi_i^t(t_1,t_2)}{w_i}-\frac{\Phi_j^t(t_1,t_2)}{w_j}\right| \le \beta\cdot\frac{\hat{L}_{max}}{w_i}+\gamma\cdot\frac{\hat{L}_{max}}{w_j},$$

where $\Phi_i^t(t_1, t_2)$ represents the transmission time used by flow i during $[t_1, t_2)$, and

$$(\beta,\gamma)=\begin{cases}\left(\frac{\hat{C}_1}{r_i^{min}}+1,\ \frac{\hat{C}_1}{r_j^{min}}+1\right), & \text{if both flows are lagging but not candidates}\\ \left(\frac{2\hat{C}_1}{r_i^{min}}+1,\ \frac{2\hat{C}_1}{r_j^{min}}+1\right), & \text{if both flows are lagging and candidates}\\ \left(\frac{2\hat{C}_1}{r_i^{min}}+1,\ \frac{2\hat{C}_1}{r_j^{min}}+1\right), & \text{if both flows are satisfied}\\ \left(\frac{(\alpha_R+1)\hat{C}_1}{r_i^{min}}+2,\ \frac{(\alpha_R+1)\hat{C}_1}{r_j^{min}}+2\right), & \text{if both flows are real-time leading flows}\\ \left(\frac{(\alpha_N+1)\hat{C}_1}{r_i^{min}}+2,\ \frac{(\alpha_N+1)\hat{C}_1}{r_j^{min}}+2\right), & \text{if both flows are non-real-time leading flows}\\ \left(\frac{\hat{C}_1}{r_i^{min}}+2,\ \frac{(2\alpha_N+1)\hat{C}_1}{r_j^{min}}+2\right), & \text{if } i \text{ and } j \text{ are real-time and non-real-time leading flows, respectively.}\end{cases}$$

3.3 Delay Bounds

Theorem 4 shows that if a lagging flow that has sufficient service demand becomes allowed-to-send and is always a candidate in the compensation scheme, it can get back all its lagging services within bounded time.

THEOREM 4. If an active but lagging flow i that remains backlogged continuously becomes allowed-to-send and is always a candidate in the compensation scheme, it is guaranteed that flow i will become nonlagging (i.e., $lag_i \le 0$) within time Δ_t, where

$$\Delta_t < \frac{\varphi(\Psi + 2\hat{L}_{max})}{w_{min}(1-\alpha_R)\hat{C}_n} + \left(\frac{\hat{C}_1}{\hat{C}_n}(m + \frac{\varphi}{w_{min}}) + 1\right)\frac{\hat{L}_{max}}{\hat{C}_n};$$

m is the number of active flows; φ, φ_R, and φ_N are the aggregate weight of all flows, all real-time flows, and all non-real-time flows, respectively; w_{min} is the minimum weight of all flows; and

$$\Psi = \begin{cases} \frac{W_R+W_N}{W_R}\left(\frac{\hat{C}_1}{\hat{C}_n}(\frac{\varphi_R \cdot lag_i(t)}{w_i} + (\frac{2\varphi_R}{w_i} + m - 2)\hat{L}_{max}) + 2\hat{L}_{max} + B\right), & \text{if flow } i \text{ is a real-time flow} \\ \frac{W_R+W_N}{W_N}\left(\frac{\hat{C}_1}{\hat{C}_n}(\frac{\varphi_N \cdot lag_i(t)}{w_i} + (\frac{2\varphi_N}{w_i} + m - 2)\hat{L}_{max}) + 2\hat{L}_{max} + B\right), & \text{if flow } i \text{ is a non-real-time flow.} \end{cases}$$

Proof. Assume that flow i is a real-time flow. Consider the worst case: flow i has the maximum lag among all flows. Since flow i becomes allowed-to-send, lag_i is never decreased after time t. Besides, because flow i is always a candidate in the compensation scheme, lag_i is decreased each time when it receives additional services. Now let $\Phi_A(t, t_N)$ be the total additional services received by all lagging flows during $[t, t+\Delta_t)$.

To prove this theorem, observe that the largest value of Δ_t occurs when all flows in the system are allowed-to-send and there is only one leading flow, say k, that provides additional services such that flow k is a real-time flow and $w_k = w_{min}$. Flow k can receive a fraction α_R of its services when it is leading, and it uses s_k to keep track of the amount of such services. So we have

$$\Phi_A(t, t+\Delta_t) \geq w_{min} \cdot \frac{\hat{C}_1}{\hat{C}_1}(v_k(t+\Delta_t) - v_k(t)) - w_{min}(s_k(t+\Delta_t) - s_k(t)) - \hat{L}_{max}. \quad (16)$$

Note that the best rate of flow k must be $\hat{C}_1$, or it is not allowed to send. By lemma 1, for any active flow j during $[t, t+\Delta_t)$, we have

$$v_j(t+\Delta_t) - v_j(t) \leq v_k(t+\Delta_t) - v_k(t) + \frac{\hat{C}_1}{r_j^{min}} \cdot \frac{\hat{L}_{max}}{w_j} + \frac{\hat{C}_1}{r_k^{min}} \cdot \frac{\hat{L}_{max}}{w_{min}} \leq v_k(t+\Delta_t) - v_k(t) + \frac{\hat{C}_1}{\hat{C}_n}\left(\frac{\hat{L}_{max}}{w_j} + \frac{\hat{L}_{max}}{w_{min}}\right).$$

This inequality helps to derive the total amount of services provided by the system during $[t, t+\Delta_t)$:

$$\begin{aligned} \hat{C}_n \cdot \Delta_t &\leq \left(\sum_{j\in A} w_j \cdot \frac{\hat{C}_1}{\hat{C}_1}(v_j(t+\Delta_t) - v_j(t))\right) + \hat{L}_{max} \\ &\leq \left(\sum_{j\in A} w_j (v_k(t+\Delta_t) - v_k(t) + \frac{\hat{C}_1}{\hat{C}_n}(\frac{\hat{L}_{max}}{w_j} + \frac{\hat{L}_{max}}{w_{min}}))\right) \\ &+ \hat{L}_{max} \leq (v_k(t+\Delta_t) - v_k(t)) \sum_{j\in A} w_j \\ &+ \frac{\hat{C}_1}{\hat{C}_n}\left(m\hat{L}_{max} + \frac{\hat{L}_{max}}{w_{min}}\sum_{j\in A} w_j\right) + \hat{L}_{max} \\ &\leq (v_k(t+\Delta_t) - v_k(t))\,\varphi \\ &+ \left(\frac{\hat{C}_1}{\hat{C}_n}(m + \frac{\varphi}{w_{min}}) + 1\right)\hat{L}_{max} \\ &\Rightarrow v_k(t+\Delta_t) - v_k(t) \\ &\geq \frac{1}{\varphi}\left(\hat{C}_n \cdot \Delta_t - (\frac{\hat{C}_1}{\hat{C}_n}(m + \frac{\varphi}{w_{min}}) + 1)\hat{L}_{max}\right). \end{aligned} \quad (17)$$

Applying lemma 4 to flow k at times t and $t+\Delta_t$ and taking a subtraction, we obtain

$$s_k(t+\Delta_t) - s_k(t) \leq \alpha_R v_k(t+\Delta_t) - \alpha_R v_k(t) + \frac{\hat{L}_{max}}{w_{min}}. \quad (18)$$

By combining equations (17) and (18) into equation (16), we can obtain

$$\begin{aligned} \Phi_A(t, t+\Delta_t) &\geq w_{min}\left(v_k(t+\Delta_t) - v_k(t) - \alpha_R v_k(t+\Delta_t) + \alpha_R v_k(t) - \frac{\hat{L}_{max}}{w_{min}}\right) - \hat{L}_{max} \\ &= w_{min}(1-\alpha_R)(v_k(t+\Delta_t) - v_k(t)) - 2\hat{L}_{max} \\ &\geq \frac{w_{min}(1-\alpha_R)}{\varphi}\left(\hat{C}_n \cdot \Delta_t - (\frac{\hat{C}_1}{\hat{C}_n}(m + \frac{\varphi}{w_{min}}) + 1)\hat{L}_{max}\right) \\ &- 2\hat{L}_{max} \Rightarrow \Delta_t \leq \frac{\varphi(\Phi_A(t, t+\Delta_t) + 2\hat{L}_{max})}{w_{min}(1-\alpha_R)\hat{C}_n} \\ &+ \left(\frac{\hat{C}_1}{\hat{C}_n}(m + \frac{\varphi}{w_{min}}) + 1\right)\frac{\hat{L}_{max}}{\hat{C}_n}. \end{aligned} \quad (19)$$

It remains to derive an upper bound for $\Phi_A(t, t+\Delta_t)$ in equation (19). The worst case happens when these $n-1$ lagging flows are candidates so that they are all allowed to share the $\Phi_A(t, t+\Delta_t)$ services. Besides, exactly one of these $n-1$ flows remains in L_N during $[t, t+\Delta_t)$. In this case, L_R can share at most a fraction $\frac{W_R}{W_R+W_N}$ of $\Phi_A(t, t+\Delta_t)$.

Let $\Phi_R(t, t+\Delta_t)$ and $\Phi_N(t, t+\Delta_t)$ be additional services received by L_R and L_N during $[t, t+\Delta_t)$, respectively, $\Phi_A(t, t+\Delta_t) = \Phi_R(t, t+\Delta_t) + \Phi_N(t, t+\Delta_t)$. By theorem 2, we have

$$\Phi_N(t, t+\Delta_t) \leq W_N \left(\frac{\Phi_R(t, t+\Delta_t)}{W_R} + \frac{B + \hat{L}_{max}}{W_R} + \frac{B + \hat{L}_{max}}{W_N} \right) \Rightarrow \Phi_A(t, t+\Delta_t) \leq \frac{W_R + W_N}{W_R} \left(\Phi_R(t, t+\Delta_t) + B + \hat{L}_{max} \right). \quad (20)$$

By applying lemma 2 twice on flow i and any flow $j \in L_R$, we have

$$\Phi_R(t, t+\Delta_t) \leq \sum_{j \in L_R} w_j \cdot \frac{\hat{C}_1}{\hat{C}_1} (c_j(t+\Delta_t) - c_j(t)) + \hat{L}_{max} \leq \sum_{j \in L_R} w_j \left(c_i(t+\Delta_t) - c_i(t) + \frac{\hat{C}_1}{r_i^{min}} \cdot \frac{\hat{L}_{max}}{w_i} + \frac{\hat{C}_1}{r_j^{min}} \cdot \frac{\hat{L}_{max}}{w_j} \right) + \hat{L}_{max} \leq (c_i(t+\Delta_t) - c_i(t)) \sum_{j \in L_R} w_j + \frac{\hat{C}_1}{\hat{C}_n} \cdot \frac{\hat{L}_{max}}{w_i} \sum_{j \in L_R} w_j + \frac{\hat{C}_1}{\hat{C}_n} \sum_{j \in L_R} \hat{L}_{max} + \hat{L}_{max} < \varphi_R (c_i(t+\Delta_t) - c_i(t)) + \left(\frac{\hat{C}_1}{\hat{C}_n} (\frac{\varphi_R}{w_i} + m - 2) + 1 \right) \hat{L}_{max}. \quad (21)$$

After time $t+\Delta_t$, flow i becomes nonlagging, so $-\hat{L}_{max} < lag(t+\Delta_t) \leq 0$. Thus, we have

$$\frac{\hat{C}_n}{\hat{C}_1} (c_i(t+\Delta_t) - c_i(t)) \leq \frac{|lag_i(t+\Delta_t) - lag_i(t)|}{w_i} < \frac{lag_i(t) + \hat{L}_{max}}{w_i} \Rightarrow c_i(t+\Delta_t) - c_i(t) < \frac{\hat{C}_1}{\hat{C}_n} \cdot \frac{lag_i(t) + \hat{L}_{max}}{w_i}. \quad (22)$$

By combining equations (21) and (22) into equation (20), we have

$$\Phi_A(t, t+\Delta_t) < \frac{W_R + W_N}{W_R} \left(\frac{\hat{C}_1}{\hat{C}_n} \left(\frac{\varphi_R \cdot lag_i(t)}{w_i} + \left(\frac{2\varphi_R}{w_i} + m - 2 \right) \hat{L}_{max} \right) + 2\hat{L}_{max} + B \right). \quad (23)$$

By combining equations (19) and (23), the first part of this theorem is proved. When flow i is a non-real-time flow, the proof is similar, and we omit the details.

4. Simulation Results

In this section, we present some experimental results to verify the effectiveness and properties of the proposed algorithm. We have developed an *event-driven* simulator by using C++ programming language. Events, such as packets' arrival and change of channel states, are tagged with timestamps and enqueued into a priority queue. The simulator then dequeues events from the priority queue and handles them by the principles of MR-FQ.

4.1 The Impact of the Multirate Environment

In the first experiment, we evaluate the impact of the multirate environment for our MR-FQ method and other wireless fair scheduling algorithms. We mix real-time and non-real-time flows together. We mainly observe the packet dropping ratios and the average queueing delays of real-time flows and the average throughput of non-real-time flows. We compare CIF-Q [10], TD-FQ [13], and the proposed MR-FQ. CIF-Q and TD-FQ are two wireless fair scheduling algorithms developed for a single-rate environment. They both assume that the wireless channel is either in a good state or a bad state. We compare MR-FQ with these two algorithms because their basic scheduling policies (i.e., Fig. 3) are similar to that of MR-FQ. (The major differences among these three scheduling algorithms are the methods of the graceful degradation scheme and compensation scheme. Besides, only MR-FQ has the rate selection scheme.) We adopt the IEEE 802.11b as the MAC protocol, which provides 11-Mb/sec, 5.5-Mb/sec, 2-Mb/sec, and 1-Mb/sec transmission rates. Ten flows are used, as shown in Table 2. The first six flows are real-time flows, which represent three traffic models: voice, video, and constant bit rate (CBR) traffics. The voice traffic is modeled as an ON-OFF process, where the average durations of ON and OFF periods are set to 2.5 and 0.5 seconds, respectively. During an ON period, packets are generated with fixed intervals. No packet is generated during an OFF period. The video traffic is modeled as variable bit rate (VBR) traffic, where packets arrive in a Poisson fashion. The last four flows are non-real-time FTP flows, and their traffic is modeled as greedy sources whose queues are never empty. The weights of these 10 flows are set to $2:1:64:32:16:8:64:64:64:64$ to reflect

Table 2. Traffic specification of the flows used in the first experiment

Flow	Guaranteed Bandwidth	Average Packet Size	Error Scenario
voice1	64 Kb/sec	2 Kb	T_{good} = 8 sec, T_{bad} = 1.5 sec
voice2	32 Kb/sec	1 Kb	T_{good} = 5 sec, T_{bad} = 1 sec
video1	2 Mb/sec	4 Kb	T_{good} = 8 sec, T_{bad} = 1.5 sec
video2	1 Mb/sec	2 Kb	T_{good} = 5 sec, T_{bad} = 1 sec
CBR1	512 Kb/sec	2 Kb	T_{good} = 8 sec, T_{bad} = 1.5 sec
CBR2	256 Kb/sec	1 Kb	T_{good} = 5 sec, T_{bad} = 1 sec
FTP1	2 Mb/sec	4 Kb	T_{good} = 9.5 sec, T_{bad} = 0.5 sec
FTP2	2 Mb/sec	4 Kb	T_{good} = 8 sec, T_{bad} = 1.5 sec
FTP3	2 Mb/sec	4 Kb	T_{good} = 5sec, T_{bad} = 1 sec
FTP4	2 Mb/sec	4 Kb	T_{good} = 3 sec, T_{bad} = 1 sec

their guaranteed bandwidth. As for error scenarios, we use two parameters, T_{good} and T_{bad}, to adjust the average time when a channel stays in good and bad states, respectively. When the channel is in the good state, the flow can use 11 Mb/sec to transmit. When the channel is in the bad state, the best transmission rate that a flow can use in MR-FQ is randomly selected from 5.5, 2, 1, and 0 Mb/sec. However, both CIF-Q and TD-FQ simply treat the channel as bad, and no packet can be transmitted. The total simulation time in this experiment is 30 minutes.

For CIF-Q, we set its parameter to $\alpha = 0.5$, while for TD-FQ and MR-FQ, we set their parameters to $\alpha_R = 0.8$ and $\alpha_N = 0.2$, respectively. In TD-FQ, the weights assigned to lagging sets are $W_R : W_N = 3 : 1$, $W_R^S : W_R^M = 3 : 1$, and $W_N^S : W_N^M = 3 : 1$. In MR-FQ, since we do not distinguish lagging flows as seriously and moderately lagging ones, there is only one ratio $W_R : W_N = 3 : 1$. Besides, the values of δ_1, δ_2, δ_3, and B in MR-FQ are set to 32, 64, 128, and 1024, respectively. Note that the units of packets are set to Kb when we compute the virtual time of flows.

The packet dropping ratios and the average queueing delays of real-time flows are shown in Figures 6 and 7, respectively, where the *packet dropping ratio* is defined as

$$\frac{\text{Number of packets dropped due to exceeding deadline}}{\text{Number of packet generated}},$$

and the deadline of a packet is set to twice of the average packet interarrival time. From Figures 6 and 7, we can observe that real-time flows have the highest packet dropping ratios and average queueing delays when we apply CIF-Q to the scheduler. This is because CIF-Q does not separate real-time flows from non-real-time flows and treat all flows in the same way. Real-time flows then have to compete with non-real-time flows, thus causing higher dropping ratios and queueing delays. The packet dropping ratios and the average queueing delays of real-time flows in TD-FQ are smaller than those in CIF-Q. This is because TD-FQ gives higher priorities to real-time flows to reduce their queueing delays (and packet dropping ratios). MR-FQ adopts the idea of TD-FQ (which gives higher priorities to real-time flows) and allows flows in a bad state to transmit packets using lower rates (if possible). So the packet dropping ratios and the average queueing delays of real-time flows in MR-FQ are smaller than those in CIF-Q and TD-FQ since the latter two methods do not allow packets to be transmitted if flows are in a bad state.

A similar effect can be observed in Figure 8, where the average throughput of non-real-time flows in MR-FQ is larger than that in CIF-Q and TD-FQ.

From this experiment, we can conclude that by considering the multirate capability of a wireless channel, the proposed MR-FQ method can reduce the packet dropping ratios and average queueing delays of real-time flows and increase the overall system performance.

4.2 The Time Fairness Property

In the second experiment, we verify the time fairness property of the MR-FQ method. Recall that there are two parts in MR-FQ that address the time fairness issue. One is the rate selection scheme, which will choose a suitable transmission rate for the selected flow according to its lagging degree and channel condition. A flow is allowed to use a lower rate for transmission only if it is suffering from seriously lagging. Another is the ratio $\frac{\hat{C}_1}{r}$, used to update a flow's virtual time (refer to equations (3), (9), and (10)), where r is the transmission rate used by the flow. To show that our MR-FQ method can satisfy the time fairness property, we design a modified version of MR-FQ that does not consider the time fairness property. This modified version removes the rate selection scheme and updates a flow i's virtual time as follows:

$$\begin{aligned} v_i &= v_i + \frac{l_p}{w_i}, \\ c_i &= c_i + \frac{l_p}{w_i}, \\ f_i &= f_i + \frac{l_p}{w_i}, \end{aligned}$$

where l_p is the length of the packet being transmitted. We mainly observe the total services received by flows and the

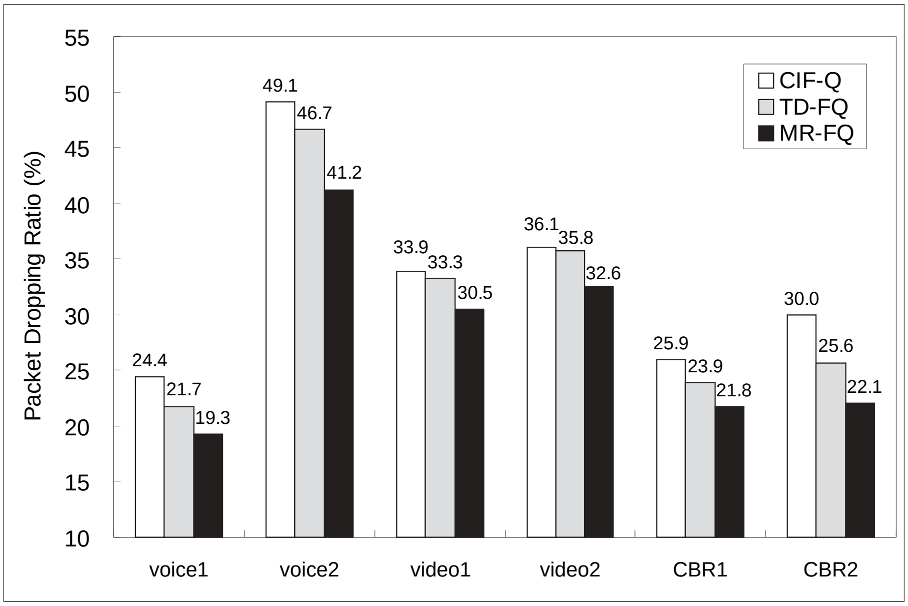


Figure 6. Packet dropping ratios of real-time flows

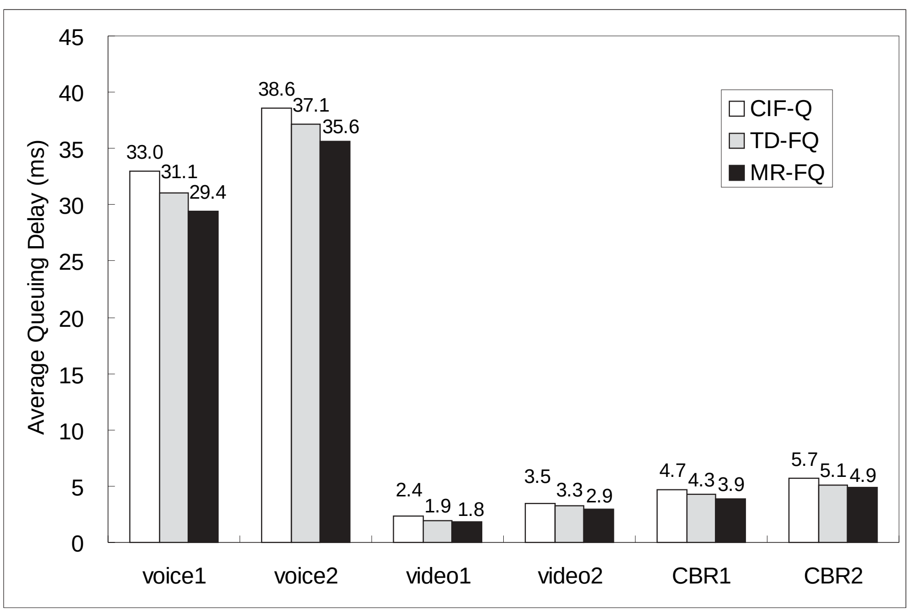


Figure 7. Average queueing delays of real-time flows

total medium time used by flows. Two FTP flows are used, as shown in Table 3. The weights of these two FTP flows are set to 1 : 1. The total simulation time in this experiment is 100 seconds.

Figures 9 and 10 show the total services received and the total medium time used by these two FTP flows, respectively. Since the channel condition of the flow FTP1 is better than that of the flow FTP2, MR-FQ will let the flow FTP1 receive more services than the flow FTP2, as shown in Figure 9(a). However, the medium time used by both flows is the same in MR-FQ, as shown in Figure 10(a). This reflects the fact that the proposed MR-FQ method can satisfy the time fairness property. On the contrary, although the modified version of MR-FQ can achieve better service fairness (as shown in Fig. 9(b)), it let the flow FTP2 occupy too much medium time, as shown in Figure 10(b).

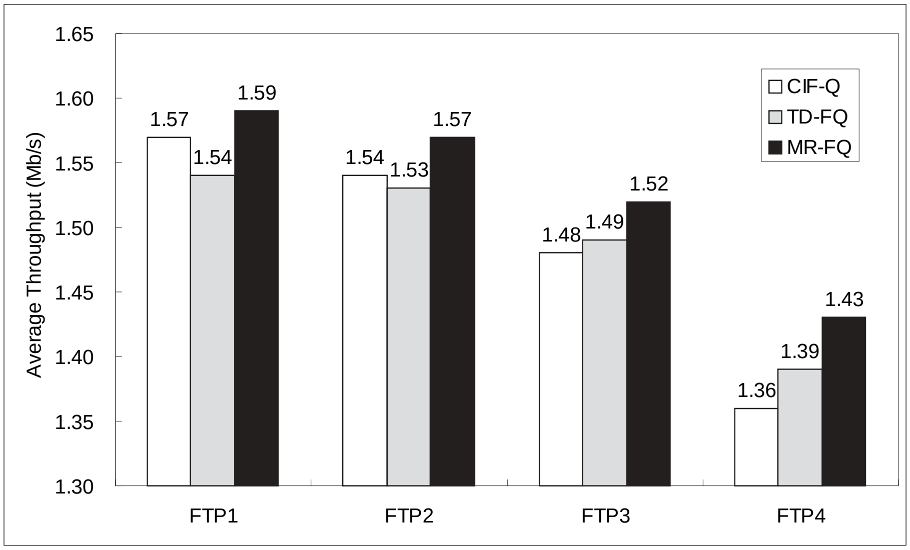


Figure 8. Average throughput of non-real-time flows

Table 3. Traffic specification of the flows used in the second experiment

Flow	Guaranteed Bandwidth	Average Packet Size	Error Scenario
FTP1	6 Mb/sec	8 Kb	T_{good} = 10 sec, T_{bad} = 1 sec
FTP2	6 Mb/sec	8 Kb	T_{good} = 4 sec, T_{bad} = 2.5 sec

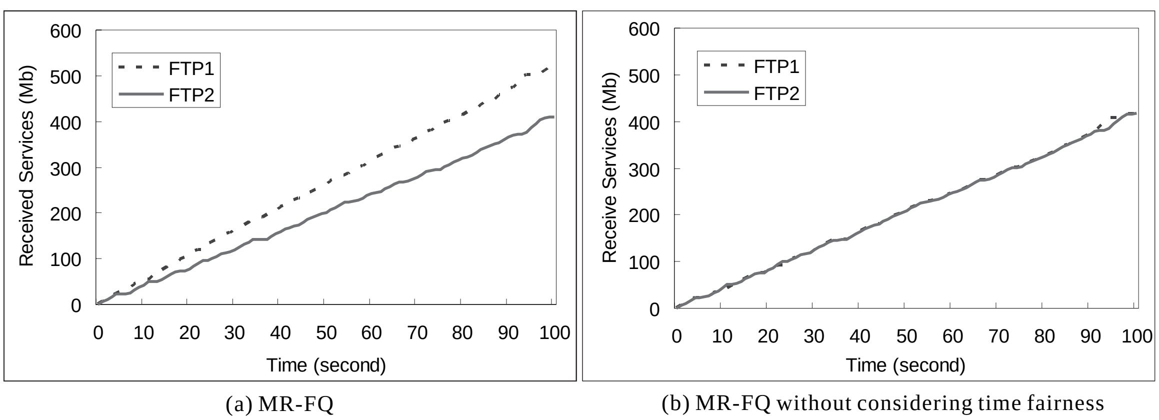
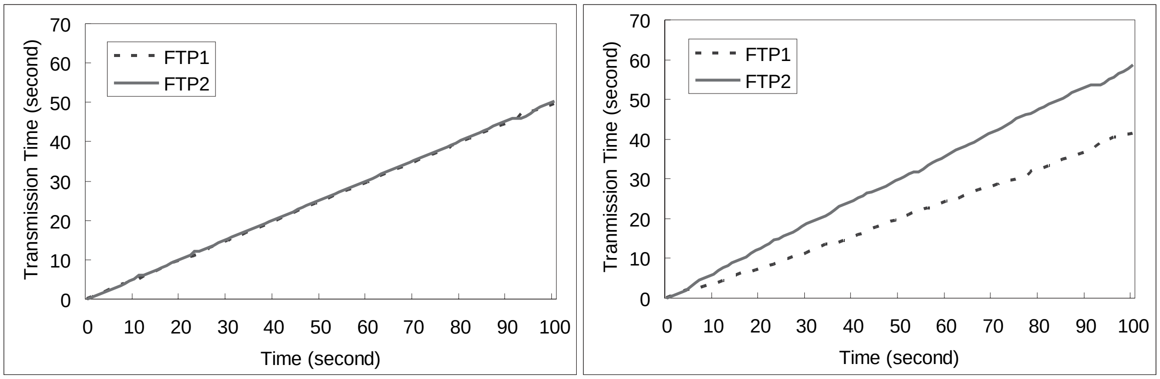


Figure 9. Total services received by the two FTP flows

(Note that since the flow FTP2 has a worse channel condition, it will often use lower transmission rates to send packets, thus causing longer transmission time.) By comparing Figure 9(a) and 9(b), we can observe that the total services received by the flow FTP1 in the modified version of MR-FQ are quite lower than that in MR-FQ. This reflects the fact that if we do not consider the time fairness issue, the flows using lower transmission rates will degrade the amount of services received by other flows (that use higher transmission rates), thus decreasing the overall system performance.

To show how bad the situation will be if we ignore the time fairness issue, we set up the third experiment. Six flows are used, as shown in Table 4. We mainly obverse



(a) MR-FQ (b) MR-FQ without considering time fairness

Figure 10. Total medium time used by the two FTP flows

Table 4. Traffic specification of the flows used in the third experiment

Flow	Guaranteed Bandwidth	Average Packet Size	Error Scenario
video1	2 Mb/sec	4 Kb	Error-free
video2	1 Mb/sec	2 Kb	T_{good} = 5 sec, T_{bad} = 3 sec
CBR1	1 Mb/sec	4 Kb	T_{good} = 10 sec, T_{bad} = 1 sec
CBR2	512 Kb/sec	2 Kb	T_{good} = 4 sec, T_{bad} = 2.5 sec
FTP1	4 Mb/sec	8 Kb	T_{good} = 9.5 sec, T_{bad} = 0.5 sec
FTP2	2 Mb/sec	4 Kb	T_{good} = 3 sec, T_{bad} = 2 sec

the services received by each flow and the total services provided by the system. The weights of these six flows are set to 4 : 2 : 2 : 1 : 8 : 4 to reflect their guaranteed rates. Other parameters used in MR-FQ are same as those in section 4.1. The total simulation time is 100 seconds.

Figures 11 and 12 show the services received by each flow and the total services provided by the system, respectively. From Figure 11, we can observe that all flows can receive more services in MR-FQ than those in the modified version of MR-FQ (which does not consider time fairness), except for the flow video2. This will imply that the total services provided by the system in MR-FQ are more than that in the modified version of MR-FQ. From this experiment, we can conclude that by considering time fairness, the proposed MR-FQ method can increase the overall system performance.

4.3 The Effect of the α_R Value on Real-Time Leading Flows

In the last experiment, we discuss the effect of different α_R values on real-time leading flows in our MR-FQ method. Recall that with the graceful degradation scheme, a real-time leading flow i can reserve approximately $\alpha_R v_i$ services. (In other words, flow i has to give up approximately $(1-\alpha_R)v_i$ services to compensate other lagging flows.) These reserved services can help reduce the queueing delays of real-time leading flows.

To evaluate the effect of different α_R values, we set up four flows, as shown in Table 5. The first three flows are real-time flows, which represent three traffic models: voice, CBR, and video traffics. The last flow is a non-real-time FTP flow. The channel conditions of these three real-time flows are much better than that of the non-real-time FTP flow. So these real-time flows will become leading flows, while the non-real-time FTP flow will become a lagging flow in this experiment. Note that the major purpose of this non-real-time FTP flow is to receive compensation services from these three real-time flows so that we can observe the effect of different α_R values on these real-time flows. The weights of these four flows are set to 1 : 8 : 32 : 64 to reflect their guaranteed bandwidth. The total simulation time in this experiment is 30 minutes. We mainly observe the packet dropping ratios (which also reflect the queueing delays) of real-time flows in this experiment.

Figure 13 shows the packet dropping ratios of these three real-time leading flows under different α_R values. The packet dropping ratios of real-time flows decrease broadly as the value of α_R increases. From Figure 13, we can observe that the α_R value does not obviously affect the packet dropping ratio of the voice flow when $\alpha_R > 0.2$. This is because the voice traffic is modeled as an ON-OFF

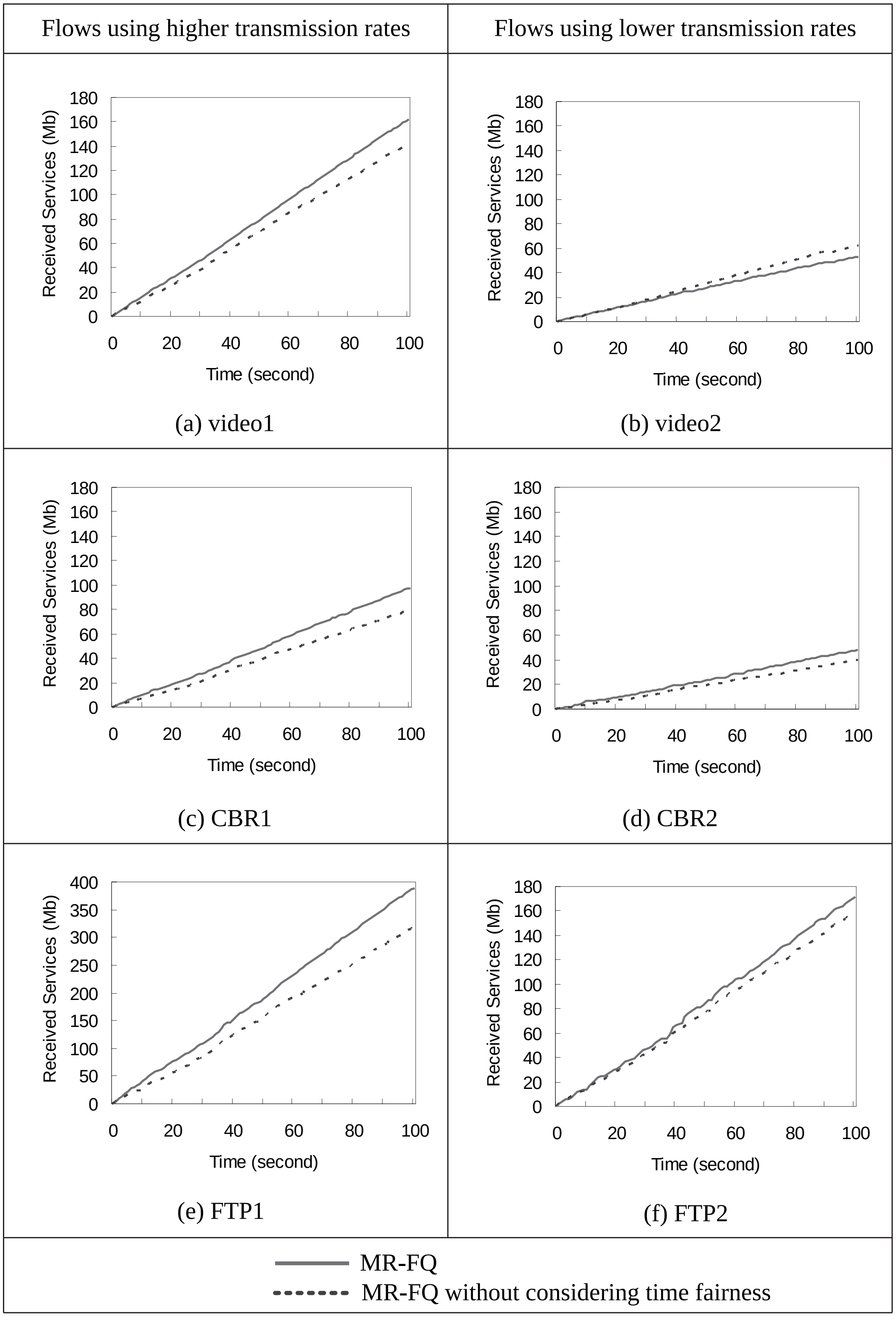


Figure 11. Total services received by each flow

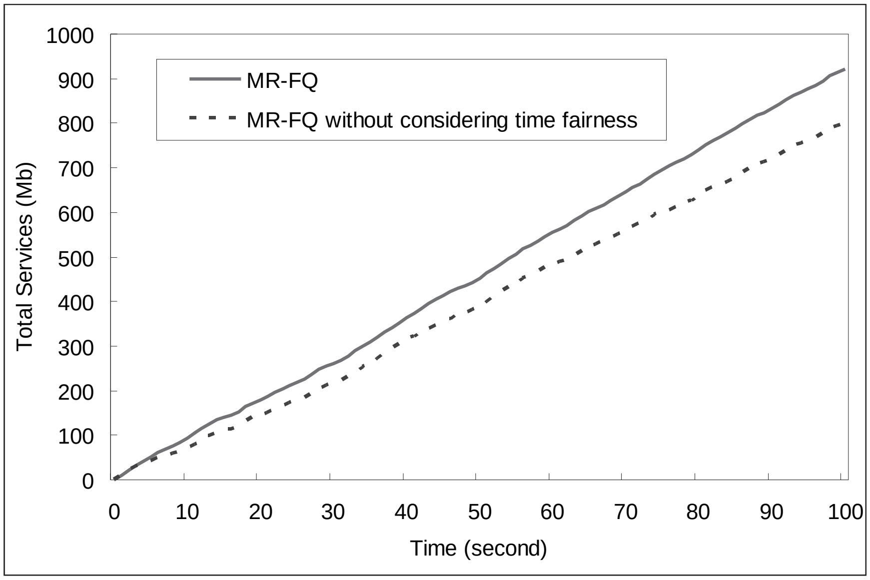


Figure 12. Total services provided by the system

Table 5. Traffic specification of the flows used in the fourth experiment

Flow	Guaranteed Bandwidth	Average Packet Size	Error Scenario
voice	64 Kb/sec	2 Kb	T_{good} = 10 sec, T_{bad} = 1 sec
CBR	512 Kb/sec	2 Kb	T_{good} = 10 sec, T_{bad} = 1 sec
video	2 Mb/sec	4 Kb	T_{good} = 10 sec, T_{bad} = 1 sec
FTP	4 Mb/sec	8 Kb	T_{good} = 3 sec, T_{bad} = 2 sec

process, and packets are generated only during an ON period. So even when we give more services to the voice flow, its queue may be empty and cannot receive such services. The packet dropping ratio of the CBR flow decreases as the value of α_R increases when $\alpha_R \leq 0.3$. This is because the packet's arrival rate is fixed in the CBR flow. When we set $\alpha_R = 0.3$ in this experiment, the CBR flow can exactly exhaust its queue content. So when $\alpha_R > 0.3$, the queue becomes empty and the packet dropping ratio of the CBR flow becomes steady. The value of α_R affects the packet dropping ratio of the video flow obviously when $\alpha_R \leq 0.6$. This is because the video flow is modeled as VBR traffic, where packets arrive in a Poisson fashion, and thus its queue may contain more packets waiting for transmission.

In summary, as we increase the value of α_R and $\alpha_R \leq \theta$, where θ is a threshold value and $\theta < 1$, the packet dropping ratios of real-time leading flows can decrease. The threshold value θ is different under various types of real-time flows. From this experiment, we can observe that $\theta_{video} > \theta_{CBR} > \theta_{voice}$, where θ_{video}, θ_{CBR}, and θ_{voice} represent the threshold values θ of video, CBR, and voice flows, respectively. Besides, as the number of flows increases, the threshold value θ also increases. This is because these real-time leading flows have to compete with more flows for transmission. If we allow them to reserve more services, then their packet dropping ratios can be reduced.

5. Conclusions

We have addressed the problem that has been ignored by many existing wireless fair scheduling algorithms that a lot of wireless networks are capable of transmitting data at multiple rates. A new algorithm, MR-FQ, is proposed to solve this problem. By taking both time fairness and service fairness into account, MR-FQ allows a flow to transmit at different rates according to its channel condition and lagging degree. It not only increases the overall system throughput but also guarantees fairness and bounded delays for flows. We have analytically derived the fairness properties and delay bounds of MR-FQ. Simulation results have also shown that MR-FQ incurs less packet dropping for real-time flows and has larger throughput for non-real-time flows when compared to CIF-Q and TD-FQ.

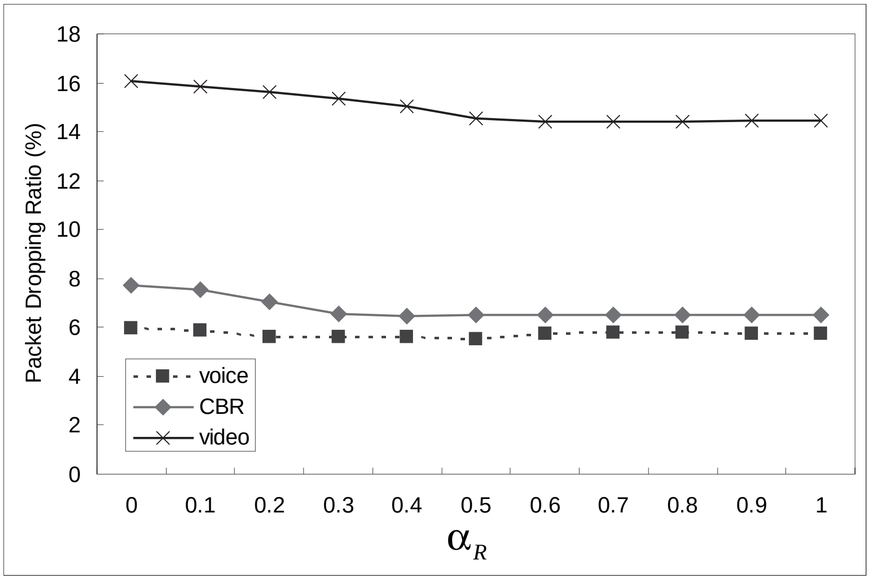


Figure 13. Packet dropping ratios of real-time flows under different α_R values

6. Appendix: Basic Lemmas

The following three lemmas give bounds on the differences between virtual times (v_is), compensation virtual times (c_is), and extra virtual times (f_is) of any two active flows.

LEMMA 1. Let $v_i(t)$ be the virtual time of flow i at time t. For any two active flows i and j such that $t \geq 0$,

$$-\frac{\hat{L}_{max}}{w_j} \times \frac{\hat{C}_1}{r_j^{min}} \leq v_i(t) - v_j(t) \leq \frac{\hat{L}_{max}}{w_i} \times \frac{\hat{C}_1}{r_i^{min}}. \quad (24)$$

Proof. This proof is by induction on t.

Basic step. When $t = 0$, all virtual times are 0, so equation (24) holds trivially.

Induction step. Suppose that at time t, equation (24) holds. Let $t + \Delta_t$ be the nearest time when any flow changes its virtual time. We want to prove equation (24) for time $t+\Delta_t$. Observe that a flow's virtual time may be updated in three cases: (1) it is selected by the scheduler and the service is indeed given to it, (2) it is selected by the scheduler but the service is given to another flow, and (3) it becomes active.

In case (1), let flow i be selected by the scheduler and use transmission rate r_i $(\geq r_i^{min})$ to send. Then its virtual time becomes $v_i(t + \Delta_t) = v_i(t) + \left(\frac{l_p}{w_i} \times \frac{\hat{C}_1}{r_i}\right)$, where l_p is the length of the packet being transmitted. By MR-FQ, it follows that $v_i(t) \leq v_j(t)$, for all $j \in A$. Since v_i is increased, by the induction hypothesis, we have

$$-\frac{\hat{L}_{max}}{w_j} \times \frac{\hat{C}_1}{r_j^{min}} \leq v_i(t + \Delta_t) - v_j(t) = v_i(t + \Delta_t) - v_j(t + \Delta_t).$$

Furthermore, since $v_i(t) \leq v_j(t)$, we have

$$v_i(t + \Delta_t) - v_j(t + \Delta_t) = \left(v_i(t) + \frac{l_p}{w_i} \times \frac{\hat{C}_1}{r_i}\right) - v_j(t) \leq \frac{\hat{L}_{max}}{w_i} \times \frac{\hat{C}_1}{r_i^{min}}.$$

So equation (24) holds at $t + \Delta_t$.

In equation (24), if flow j is selected by the scheduler and uses transmission rate r_j $(\geq r_j^{min})$ to send, then $v_i(t + \Delta_t) - v_j(t+\Delta_t) \leq \frac{\hat{L}_{max}}{w_i} \times \frac{\hat{C}_1}{r_i^{min}}$ holds trivially. Furthermore,

$$v_i(t + \Delta_t) - v_j(t + \Delta_t) = v_i(t) - \left(v_j(t) + \frac{l_p}{w_j} \times \frac{\hat{C}_1}{r_j}\right) \geq -\frac{\hat{L}_{max}}{w_j} \times \frac{\hat{C}_1}{r_j^{min}}.$$

So equation (24) still holds at $t + \Delta_t$.

Case (2) is similar to case (1), except that we need to replace r_i and r_j by $\hat{C}_1$ in all inequalities.

In case (3), suppose that flow i becomes active at $t+\Delta_t$. By MR-FQ, $v_i(t+\Delta_t)$ is set to $max\{v_i(t), min_{k\in A-\{i\}}\{v_k(t+\Delta_t)\}\}$. If $v_i(t+\Delta_t) = min_{k\in A-\{i\}}\{v_k(t+\Delta_t)\}$, then equation (24) holds trivially. Otherwise, $v_i(t+\Delta_t) = v_i(t)$, which means that $v_i(t) \geq min_{k\in A-\{i\}}\{v_k(t+\Delta_t)\}$. So we have

$$v_i(t+\Delta_t) - v_j(t+\Delta_t) \geq min_{k\in A-\{i\}}\{v_k(t+\Delta_t)\}$$

$$- v_j(t+\Delta_t) \geq -\frac{\hat{L}_{max}}{w_j} \times \frac{\hat{C}_1}{r_j^{min}}.$$

Since the virtual time is nondecreasing, we have

$$v_i(t+\Delta_t) - v_j(t+\Delta_t) \leq v_i(t) - v_j(t) \leq \frac{\hat{L}_{max}}{w_i} \times \frac{\hat{C}_1}{r_i^{min}}.$$

So equation (24) holds at $t+\Delta_t$. When flow j becomes active, the proof is similar, so we can conclude the proof.

Since MR-FQ updates c_i and f_i similarly to that of v_i, proofs of the next two lemmas are similar to that of lemma 1. So we omit the proofs.

Lemma 2. Let $c_i(t)$ be the compensation virtual time of flow i at time t. For any two flows i and j that are both candidates and have the same traffic type (real-time or non-real-time flows) such that $t \geq 0$, we have

$$-\frac{\hat{L}_{max}}{w_j} \times \frac{\hat{C}_1}{r_j^{min}} \leq c_i(t) - c_j(t) \leq \frac{\hat{L}_{max}}{w_i} \times \frac{\hat{C}_1}{r_i^{min}}.$$

Lemma 3. Let $f_i(t)$ be the extra virtual time of flow i at time t. For any two flows i and j that are both candidates such that $t \geq 0$, we have

$$-\frac{\hat{L}_{max}}{w_j} \times \frac{\hat{C}_1}{r_j^{min}} \leq f_i(t) - f_j(t) \leq \frac{\hat{L}_{max}}{w_i} \times \frac{\hat{C}_1}{r_i^{min}}.$$

The next lemma gives bounds on the difference between the normalized services received by a leading flow i (i.e., s_i) and the maximum amount that it can receive (i.e., $\alpha_i v_i$).

Lemma 4. Let $s_i(t)$ be the value of s_i at time t. For any flow i that is allowed-to-send, backlogged, and leading during the time interval $t \in [t_1, t_2)$, we have

$$(\alpha - 1)\frac{\hat{L}_{max}}{w_i} \leq \alpha v_i(t) - s_i(t) \leq \alpha \frac{\hat{L}_{max}}{w_i}, \qquad (25)$$

where $\alpha = \alpha_R$ if flow i is a real-time flow, and $\alpha = \alpha_N$ otherwise.

Proof. The proof is by induction on time $t \in [t_1, t_2)$.

Basic step. When $t = t_1$, flow i just becomes leading, so the graceful degradation scheme sets $s_i(t) = \alpha v_i(t)$, and the lemma is trivially true.

Induction step. Suppose that at time t, the lemma holds. Observe that v_i and/or s_i change only when flow i is selected. So we consider two cases: (1) flow i is actually served, and (2) another flow $j \neq i$ is served. Let $t+\Delta_t \leq t_2$ be the nearest time that v_i and/or s_i are updated. We prove that the lemma still holds at $t+\Delta_t$.

According to MR-FQ, case (1) occurs only when $s_i(t) \leq \alpha v_i(t)$, so we have

$$\alpha v_i(t+\Delta_t) - s_i(t+\Delta_t) = \alpha\left(v_i(t) + \frac{l_p}{w_i}\right)$$

$$-\left(s_i(t) + \frac{l_p}{w_i}\right) = (\alpha - 1)\frac{l_p}{w_i} + \alpha v_i(t)$$

$$- s_i(t) \geq (\alpha - 1)\frac{\hat{L}_{max}}{w_i},$$

where l_p represents the length of the packet being transmitted.

Case (2) implies $s_i(t) > \alpha v_i(t)$. Also, v_i is updated, but s_i is not. So we have

$$\alpha v_i(t+\Delta_t) - s_i(t+\Delta_t) = \alpha(v_i(t) + \frac{l_p}{w_i})$$

$$- s_i(t) < \alpha\frac{l_p}{w_i} \leq \alpha\frac{\hat{L}_{max}}{w_i}.$$

Lemma 5. Let $V_R(t)$ and $V_N(t)$ be the value of V_R and V_N, respectively. For $t \geq 0$, we have

$$-\frac{B}{W_N} \leq V_R(t) - V_N(t) \leq \frac{B}{W_R}.$$

Proof. This proof is by induction on time $t \geq 0$.

Basic step. When $t = 0$, $V_R(t) = V_N(t) = 0$, so the lemma is trivially true.

Induction step. Assume that the lemma holds at time t. V_R (respectively, V_N) can be updated only when L_R^k (respectively, L_N^k) is nonempty, where L_R^k (respectively, L_N^k) is the subset of L_R (respectively, L_N) selected in the compensation scheme, respectively. We consider two cases: (1) only one set is nonempty, and (2) two sets are nonempty. Let $t+\Delta_t$ be the nearest time that V_R or V_N is updated. We want to prove the lemma to be true at time $t+\Delta_t$.

In case (1), if L_R^k is nonempty, additional services are given to L_R. In MR-FQ, we bound the total difference of additional services received by L_R and L_N at any time by $|W_R V_R - W_N V_N| \leq B$. So at time $t+\Delta_t$, $W_R V_R(t+\Delta_t) - W_N V_N(t+\Delta_t) \leq B$. Since $W_R \geq W_N$, we have

$$W_R V_R(t+\Delta_t) - W_R V_N(t+\Delta_t) \leq W_R V_R(t+\Delta_t)$$

$$- W_N V_N(t+\Delta_t) \leq B \Rightarrow V_R(t+\Delta_t)$$

$$- V_N(t+\Delta_t) \leq \frac{B}{W_R}.$$

On the other hand, if L_N^k is nonempty, we can similarly derive that $V_R(t+\Delta_t) - V_N(t+\Delta_t) \geq -\frac{B}{W_N}$. So the lemma holds at $t+\Delta_t$.

In case (2), since both sets are nonempty, the scheduler gives additional services to L_R if $V_R(t) \leq V_N(t)$. Let l_p represent the length of the packet being transmitted. We have

$$V_R(t+\Delta_t) - V_N(t+\Delta_t) = \left(V_R(t) + \frac{l_p}{W_R}\right) - V_N(t) \leq \frac{l_p}{W_R} \leq \frac{\hat{L}_{max}}{W_R} \leq \frac{B}{W_R}.$$

Note that it is trivially true that $-\frac{B}{W_N} \leq V_R(t+\Delta_t) - V_N(t+\Delta_t)$. Similarly, if $V_R(t) > V_N(t)$, the service is given to L_N, so we have

$$V_R(t+\Delta_t) - V_N(t+\Delta_t) = V_R(t) - \left(V_N(t) + \frac{l_p}{W_N}\right) > -\frac{l_p}{W_N} \geq -\frac{\hat{L}_{max}}{W_N} \geq -\frac{B}{W_N}.$$

Note that it is trivially true that $V_R(t+\Delta_t) - V_N(t+\Delta_t) \leq \frac{B}{W_R}$. Therefore, the lemma still holds at $t + \Delta_t$.

7. Acknowledgments

Y. C. Tseng's research is cosponsored by the NSC Program for Promoting Academic Excellence of Universities under grant 93-2752-E-007-001-PAE, by Computer and Communications Research Labatories (ITRI, Taiwan), by the Communications Software Technology Project of III (Taiwan), and by Intel, Inc.

8. References

[1] Parekh, A. K., and R. G. Gallager. 1993. A generalized processor sharing approach to flow control in integrated services networks: The single-node case. *IEEE/ACM Transactions on Networking* 1 (3): 344-57.

[2] Demers, A., S. Keshav, and S. Shenker. 1990. Analysis and simulation of a fair queueing algorithm. *Journal of Internetworking Research and Experience* 1:3-26.

[3] Goyal, P., H. M. Vin, and H. Chen. 1996. Start-time fair queueing: A scheduling algorithm for integrated services packet switching networks. In *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 157-68.

[4] Golestani, S. J. 1994. A self-clocked fair queueing scheme for broadband applications. *INFOCOM*, pp. 12-6.

[5] Bennett, J. C. R., and H. Zhang. 1996. $\mathcal{W}F^2Q$: Worst-case fair weighted fair queueing. *INFOCOM* 1:120-8.

[6] Bennett, J. C. R., and H. Zhang. 1996. Hierarchical packet fair queueing algorithms. In *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 143-56.

[7] Bharghavan, V., S. Lu, and T. Nandagopal. 1999. Fair queueing in wireless networks: Issues and approaches. *IEEE Personal Communications* 6:44-53.

[8] Cao, Y., and V. O. K. Li. 2001. Scheduling algorithms in broadband wireless networks. In *Proceedings of IEEE* 89 (1): 76-87.

[9] Lu, S., V. Bharghavan, and R. Srikant. 1999. Fair scheduling in wireless packet networks. *IEEE/ACM Transactions on Networking* 7 (4): 473-489.

[10] Ng, T. S. E., I. Stoica, and H. Zhang. 1998. Packet fair queueing algorithms for wireless networks with location-dependent errors. *INFOCOM*, vol. 3, 1103-11.

[11] Ramanathan, P., and P. Agrawal. 1998. Adapting packet fair queueing algorithms to wireless networks. *ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 1-9.

[12] Lu, S., T. Nandagopal, and V. Bharghavan. 1998. A wireless fair service algorithm for packet cellular networks. In *ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 10-20.

[13] Wang, Y. C., S. R. Ye, and Y. C. Tseng. 2005. A fair scheduling algorithm with traffic classification in wireless networks. *Computer Communications* 28 (10): 1225-39.

[14] LAN/MAN Standards Committee of the IEEE Computer Society. 1999. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: High-speed physical layer in the 5 GHz band, IEEE Std. 802.11a.

[15] LAN/MAN Standards Committee of the IEEE Computer Society. 1999. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Higher-speed physical layer extension in the 2.4 GHz band, IEEE Std. 802.11b.

[16] Sadeghi, B., V. Kanodia, A. Sabharwal, and E. Knightly. 2002. Opportunistic media access for multirate ad hoc networks. In *International Conference on Mobile Computing and Networking*, pp. 24-35.

[17] Eckhardt, D. A., and P. Steenkiste. 2000. Effort-limited fair (ELF) scheduling for wireless networks. In *INFOCOM*, vol. 3, 1097-1106.

[18] Lin, P., B. Benssou, Q. L. Ding, and K. C. Chua. 2000. CS-WFQ: A wireless fair scheduling algorithm for error-prone wireless channels. In *Proceedings of Computer Communications and Networks*, pp. 276-81.

[19] Wang, L., Y. K. Kwok, W. C. Lau, and V. K. N. Lau. 2002. Channel capacity fair queueing in wireless networks: Issues and a new algorithm. In *IEEE International Conference on Communications*, pp. 3116-20.

[20] Stamoulis, A., and G. B. Giannakis. 2000. Packet fair queueing scheduling based on multirate multipath-transparent CDMA for wireless networks. In *INFOCOM*, vol. 3, pp. 1067-76.

[21] Subramanian, S., and T. F. Wong. 1999. Transmission scheduling scheme for multirate multicode CDMA networks. In *MILCOM*, vol. 1, pp. 403-7.

[22] Yoon, J. H., M. J. Sheen, and S. C. Park. 2003. Scheduling methods with transmit power constraint for CDMA packet services. *VTC Spring* 2:1450-3.

[23] Stamoulis, A., N. D. Sidiropoulos, and G. B. Giannakis. 2004. Time-varying fair queueing scheduling for multicode CDMA based on dynamic programming. *IEEE Transactions on Wireless Communications* 3:512-23.

[24] Ue, T., S. Sampei, N. Morinaga, and K. Hamaguchi. 1998. Symbol rate and modulation level-controlled adaptive modulation/TDMA/TDD system for high-bit-rate wireless data transmission. *IEEE Transactions on Vehicular Technology* 47:1134-47.

[25] Balachandran, K., S. R. Kadaba, and S. Nanda. 1999. Channel quality estimation and rate adaptation for cellular mobile radio. *IEEE Journal on Selected Areas in Communications* 17:1244-56.

[26] Holland, G., N. Vaidya, and P. Bahl. 2001. A rate-adaptive MAC protocol for multi-hop wireless networks. In *International Conference on Mobile Computing and Networking*, pp. 236-51.

[27] Yi, Y., Y. Seok, T. Kwon, Y. Choi, and J. Park. 2000. $\mathcal{W}^2F^2Q$: Packet fair queueing in wireless packet networks. In *ACM International Workshop on Wireless Mobile Multimedia*, pp. 2-10.

You-Chiun Wang *received his B.S. and M.S. degrees in computer science from National Chung-Cheng University and National Chiao-Tung University, Taiwan, in 2001 and 2003, respectively.*

He is currently a Ph.D. student of the Department of Computer Science and Information Engineering, National Chiao-Tung University. His research interests include mobile computing and wireless communication.

Yu-Chee Tseng *is a full professor at the Department of Computer Science and Information Engineering, National Chiao-Tung University, Taiwan. He served as a program chair in the Wireless Networks and Mobile Computing Workshop (2000 and 2001); as a vice program chair at the International Conference on Distributed Computing Systems (ICDCS, 2004); as a vice program chair at the IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS, 2004); as an associate editor for* The Computer Journal*; as a guest editor for the* ACM Wireless Networks *special issue on "Advances in Mobile and Wireless Systems"; as a guest editor for the* IEEE Transactions on Computers *special issue on "Wireless Internet"; as a guest editor for the* Journal of Internet Technology *special issue on "Wireless Internet: Applications and Systems"; as a guest editor for the* Wireless Communications and Mobile Computing *special issue on "Research in Ad Hoc Networking, Smart Sensing, and Pervasive Computing"; as an editor for the* Journal of Information Science and Engineering*; as a guest editor for the* Telecommunication Systems *special issue on ''Wireless Sensor Networks"; and as a guest editor for the* Journal of Information Science and Engineering *special issue on "Mobile Computing." He is a two-time recipient of the Outstanding Research Award, National Science Council, ROC, in 2001-2002 and 2003-2005, and a recipient of the Best Paper Award at the International Conference on Parallel Processing, 2003.*

Wen-Tsuen Chen *has been with the National Tsing Hua University since 1976 and is currently a chair professor of the Department of Computer Science. He has served as department chairman, dean of the College of Electrical Engineering & Computer Science, and director of the Science & Technology Advisory Office, Ministry of Education, Taiwan. He has consulted at various levels of the Taiwan government and served as a member of many planning and technical review boards. For 14 years, he has served as co-chairman and chairman of the Technical Evaluation Board of the Ministry of Economic Affairs for Promoting High-Tech Products and Technologies, which is recognized as most pivotal in promoting industrial technologies in Taiwan. He pioneered the design of computer networks and parallel systems in early 1980s. He is currently leading a project for design and applications of advanced information networks. He has received numerous awards for his achievements in computer networking and parallel processing, including Outstanding Research Awards of the National Science Council, National Chair of the Ministry of Education, and Technical Achievement Award of the IEEE Computer Society. Professor Chen was the general chair of the 2000 IEEE International Conference on Distributed Computing Systems and the founding general chair of the IEEE International Conference on Parallel and Distributed Systems. He is an IEEE fellow and a fellow of the Chinese Society for Management of Technology.*